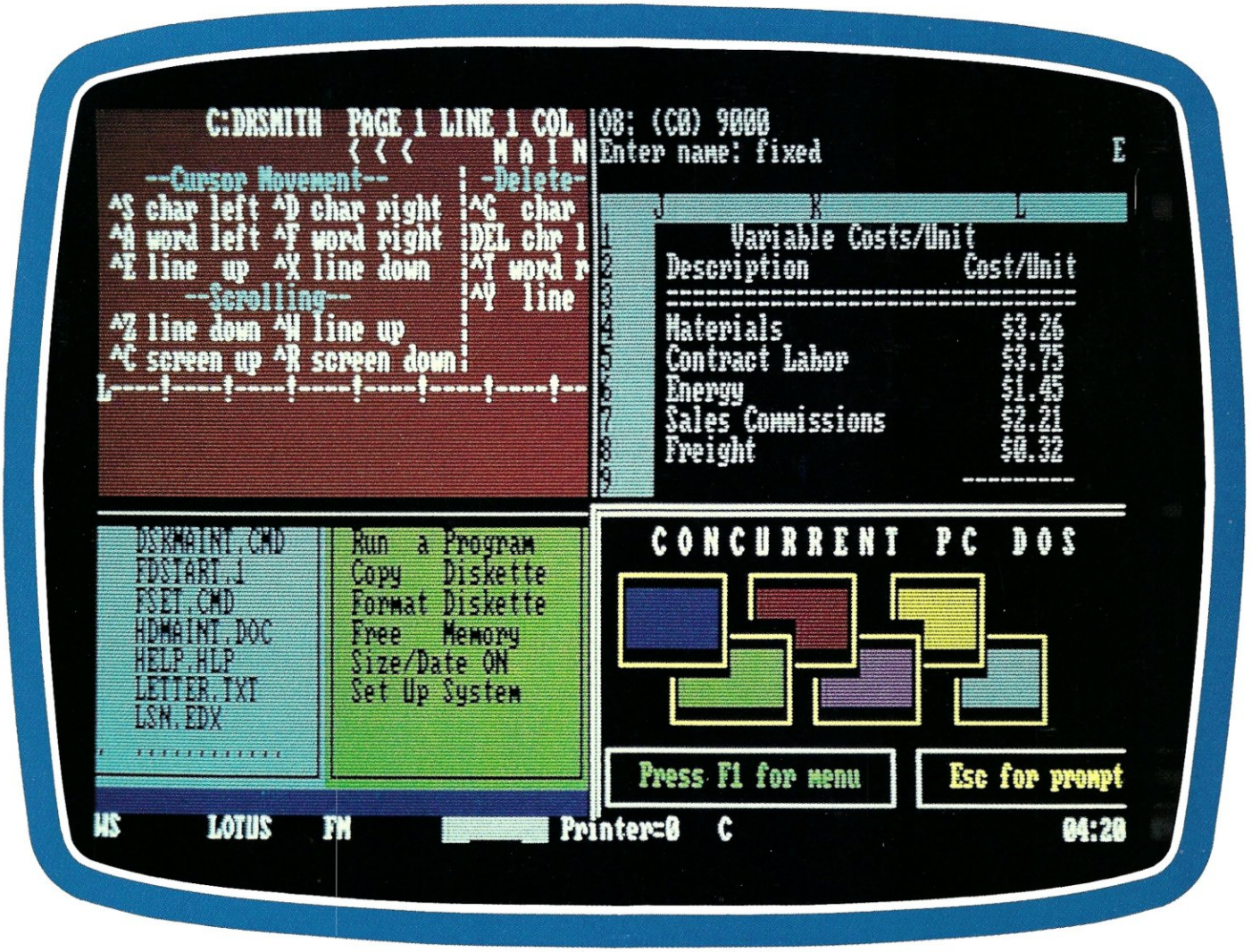
Vol. 3 No. 3 May/June 1987 $3.95 USA

Micro/Systems Journal™

For the Advanced Computer User



Multitasking/Multiuser Operating Systems

see pages 22-46

Also in this Issue:

Set Your Computer Clock to a National Standard	12
Using Turbo Pascal's Chain Function	19
Buying an AT Clone System	58
Number Crunching on a Sun 3/260 Workstation	65
Cabling for LAN Systems	74

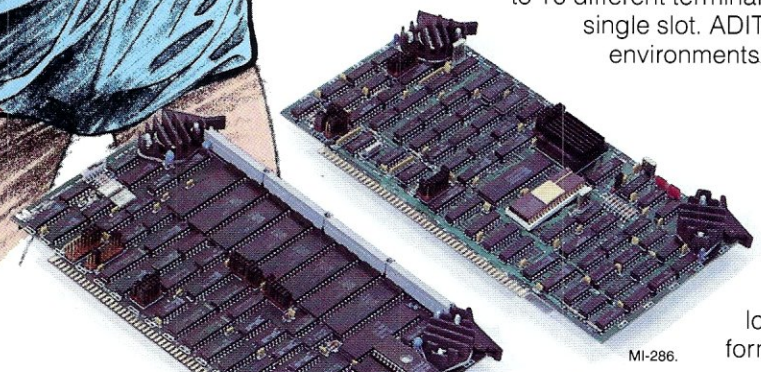
05
38351 19902 9

MACROTECH—STILL THE S-100 PERFORMANCE PACESETTER

MI-286. Our 80286/Z80H Dual CPU Board is at least twice as fast as Compupro's 8085/88 and it's a direct replacement. The MI-286 has already become the standard by which other 80286 based systems are measured. Ask us for a complimentary Benchmark Report.

ADIT. There's nothing else like it on the market. It's an Intelligent I/O Board with its own real time firmware that lets you control up to 16 different terminals, modems or printers all from a single slot. ADIT is the performance standard in environments such as Alpha Micro where I/O speed is critical.

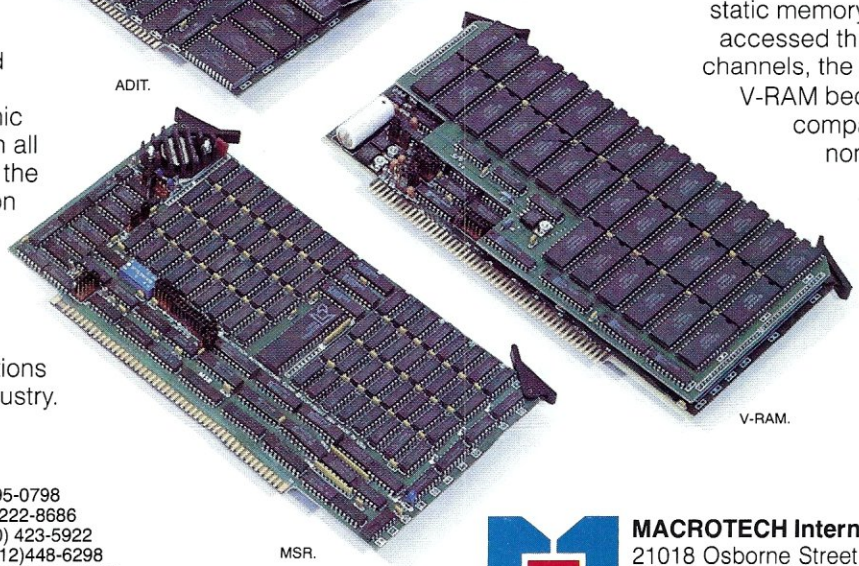
V-RAM. High performance Static CMOS system memory/virtual disk in either quarter or half megabyte configurations. With its on-board battery and power-fail logic, the V- RAM sets a new performance standard at conventional static memory prices. When accessed through I/O port channels, the half megabyte V-RAM becomes M Drive compatible with true non-volatile solid-state disk capability.



MI-286.

ADIT.

MSR. High performance and reliability in a memory so fast you won't believe it's a dynamic ram product. Compatible with all popular S-100 environments, the MSR's low power consumption and 120 nanosecond ram devices set a new standard for dynamic memory products. The MSR is available in quarter, half, one and two megabyte configurations at the lowest prices in the industry.



V-RAM.

MSR.

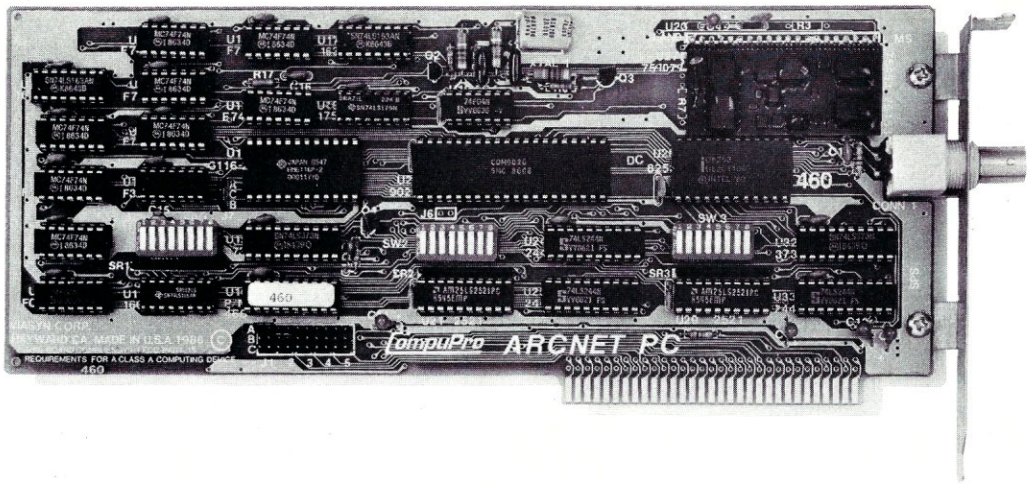
Dealers:
Gifford Computer Systems (415) 895-0798
Custom Computer Technology (800) 222-8686
S-100 (800) 423-5922
John D. Owens & Associates (212)448-6298
In England; Fulcrum (Europe) Ltd. (0621) 828763

Macrotech dealers also include most Compupro Systems Centers, Heathkit Electronic Centers and Alpha Micro Dealers.

MACROTECH International Corp.
21018 Osborne Street
Canoga Park, CA 91304
(800) 824-3181 • in Calif. (818) 700-1501
Telex: 9109970653

The Better Net

The CompuPro ARCNET® PC Board



Better Compatibility

The **CompuPro ARCNET PC** is compatible with all versions of Novell Netware and Advanced Netware™, *and* DR-NET™ running in all types of PC-compatibles. No other board can say that!

Better Reliability

The **CompuPro ARCNET PC** board offers distinctly more reliability than other boards. Why? Read on.

Better Design

The **CompuPro ARCNET PC** board uses unique circuit design to increase signal margins. Better interface circuitry insures operation with even the fastest PC-compatibles.

Better Specs

The **CompuPro ARCNET PC** board meets all FCC requirements!

Better Made

The **CompuPro ARCNET PC** board is manufactured, tested and burned in right here in the USA. It is not subject to the wide swings in quality often experienced with foreign-manufactured boards.

Better Value

All of this adds up to the fact that the **CompuPro ARCNET PC** board is the best value in Novell compatible network boards.

Better Call Today

Call us today for the name of your nearest **CompuPro** dealer. They are prepared to discuss a variety of network and multi-user solutions to meet your requirements.

CompuPro™

Viasyn Corporation
26538 Danti Court
Hayward, CA 94545-3999

(415) 786-0909
TWX: 510-100-3288
Easylink Mailbox 62877579

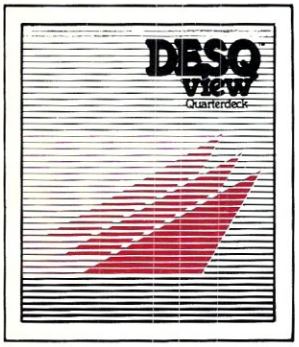
Trademarks: CompuPro: Viasyn Corporation. ARCNET: Datapoint Corporation. Netware, Advanced Netware: Novell, Inc. DR-NET: Digital Research Inc.

INFO WORLD 1986
PRODUCT OF THE YEAR
READERS' CHOICE

Window shoppers. Check out DESQview.™

Thru its windows, see tomorrow today.

In 1984, Quarterdeck created a new class of IBM PC software—the multi-window operating environment. DESQview. There was nothing quite like it then and there still isn't.



Power packed.

DESQview gives your PC the power of 9 PC's and then some. It does windows. It multi-tasks. It lets your favorite programs (including graphics software) pop up when you need them. It transfers data. It dials your phone. It makes DOS easy. It remembers your keystrokes (macros). And lots more.

The secret is out.

DESQview is the window choice of the discriminating shopper. The shopper looking for performance. Quality. The best value.

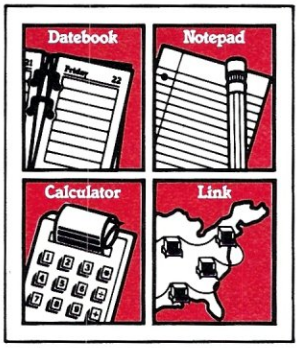
DESQview ubiquitous.

Because over 250,000 DESQviews are out in the marketplace, you'll find DESQview all over the USA; in Europe and even in Brazil. It is used in Fortune 500 corporations and in small businesses. It is used by power PC users and novices. You'll find DESQview running not just with 1-2-3, dBASEIII, WordPerfect and other popular business software, but also with telephone answering devices, video cameras, music synthesizers. Running on networks. In 3270 links to mainframes and with radio broadcast stock quotation systems. And on 386 PC's.

But that's not all.

Quarterdeck is introducing **DESQview Companions.**™ A set of four programs. **DESQview Datebook.**™ **DESQview Notepad.**™ **DESQview Calculator.**™ **DESQview Link.**™ All of them friends to each other yet extremely powerful each in its own way.

Check out **Datebook's** day, week, month, year at a glance, its alarms and its snooze. Check out **Notepad's** editing acumen. Check out **Calculator's** number prowess. Check out **Link's** electronic mail and automatic communications. But above all, check out DESQview. See its windows...learn the vision...feel its power ...and as time passes enhance your excellence with its. For DESQview windows into tomorrow today (and its **Companions** could be yours for many years to come).



Yes! I want to check out DESQview and its Companions! Today!

Copies	Product	Price ea.	Total
_____	**DESQview**	$ 99.95	$_____
_____	**Companions Only**	99.95*	_____
_____	**DESQview & Companions**	174.95**	_____
	Shipping & Handling–USA	6.00	
	Outside USA	10.00	_____
	Sales Tax (CA Residents)	6.5%	_____
	Amount enclosed		$_____

10 day money back guarantee

To order or for a dealer nearest you, call (213) 392-9851

Payment: ☐ VISA ☐ MC ☐ AMEX ☐ CHECK

Credit card: Valid Since ____/____ Expiration ____/____

Card #: ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Name _____

Shipping address _____

_____

City _____ **State** _____ **Zip** _____

Telephone _____

*If you purchased DESQview or received DESQview bundled with a multi-function board or a PC, call us about our special registered user "thank-you" bonus.

**Limited time offer until April 1, 1987.

Minimum system requirements: IBM PC, XT, AT or 100% compatible; 512K memory required

DESQ™ view Quarterdeck

Quarterdeck Office Systems
150 Pico Boulevard
Santa Monica, CA 90405
(213) 392-9851

MSJ587

DESQview, DESQview Companions, DESQview Notepad, DESQview Datebook, DESQview Calculator and DESQview Link, are trademarks of Quarterdeck Office Systems; IBM is a registered trademark of International Business Machines Corporation; 1-2-3 is a registered trademark of Lotus Development Corporation; dBASEIII is a registered trademark of Ashton Tate; WordPerfect is a registered trademark of SSI Software.

May/June 1987 Vol. 3. No. 3

Micro/Systems Journal™

For the Advanced Computer User

About the cover: *Concurrent PC DOS from Digital Research is one of four multitasking/multiuser operating systems reviewed this month. Here, Lotus 1-2-3, Wordstar, File Manager and the Menu System are shown running simultaneously under Concurrent PC DOS.*

Feature Articles

Buying an AT Clone System *by Sol Libes* **58**
To assemble, or not to assemble, an AT clone—that is the question.

Asking Questions and Getting Responses in an MS-DOS Batch File *by Bill Rogers* **62**
Creating menus and taking alternate actions in MS-DOS Batch files.

Program Interfacing to Microsoft Windows *by William Wong* **48**
Part III—A look at functions used to create and manipulate windows within Microsoft Windows.

Product Reviews

Reviews of four multitasking/multiuser operating systems for the IBM PC, PC/XT, PC/AT and compatibles.

Concurrent DOS 5.0 XM *by Alex K. H. Soya* **22**

Santa Cruz XENIX V *by Wynford Flan* **30**

Wendin's Operating System Toolbox, PCVMS & PCNX *by Stephen Randy Davis* **36**

VENIX System V *by Jim Rosenberg* **42**

Columns

The C Forum *by Don Libes* **12**
Setting your computer's clock to a national reference, the Naval Observatory in Washington, D.C.

Turbo Pascal Corner *by Stephen Randy Davis* **19**
Using the Turbo Pascal Chain Files function to save executable programs in substantially less disk space.

The Scientific Computer User *by A.G.W. Cameron* **65**
Number crunching on a Sun 3/260 workstation—frustration and elation.

The UNIX File *by Ian F. Darwin* **68**
A review of MINIX, a UNIX clone with full source code. And, how to roll your own AT&T 3B computer from a Single Board Computer.

The Public Domain Software Forum *by Charles H. Strom* **72**
The author sings the praises of CED, a public domain DOS Command Editor program and Referee, a low-cost Terminate-and-Stay-Resident program that manages other TSR's.

LANScape *by B.J. Hall and Michael Cherry* **74**
Part II—A look at cabling systems used to interconnect devices on the local-area network.

Departments

Editor's Page *by Sol and Don Libes* **4**

News, Views & Gossip *by Sol Libes* **6**

Mail **10**

New Products **76**

The Software Directory **79**

Editorial
Founder and Editor *Sol Libes*
Technical Editor *Don Libes*
Associate Editors *Lennie Libes*, *Susan Libes*
Contributing Editors *Robert A. Blum*, *A.G.W. Cameron*, *Michael Cherry*, *Ian F. Darwin*, *Stephen R. Davis*, *B.J. Hall*, *Stephen M. Leon*, *Charles H. Strom*
Consulting Editor *Michael Swaine*
Editorial Coordinator *Kobi Morgan*
Copy Editor *Rhoda Simmons*

Production
Production Manager *Bob Wynne*
Art Director *Kate Paddock*
Production *Kobi Morgan*
Typesetter *Lorraine Buckland*

Circulation
Director of Circulation *Maureen Kaminski*
Newsstand Sales Mgr. *Stephanie Barber*
Circulation Coordinator *Kathleen Shay*
Circulation Assistant *Sarah Frisbie*

Administration
Finance Manager *Kate Wheat*
Business Manager *Betty Trickett*
Accounts Payable Supv. *Mayda Lopez-Quintana*
Accts. Pay. Coordinator *Kathy Robinson*
Accounts Payable Asst. *Patricia Albert*
Accounts Receivable Mgr. *Laura Di Lazzaro*
Accts. Receivable Asst. *Denise Giannini*

Advertising
Advertising Director *Richard Mixter*
National Account Mgr. *Tami Beaulieu (213) 470-5959*
Northwestern Account Mgr. *Patricia Bacci (415) 366-3600*
Advertising Coordinator *Charles Shively*

M & T Publishing, Inc.
Chairman of the Board *Otmar Weber*
Director *C. F. von Quadt*
President and Publisher *Laird Foshay*

Micro/Systems Journal (ISSN #8750-9482) is published bimonthly by M & T Publishing, Inc., 501 Galveston Dr., Redwood City, CA 94063; (415) 366-3600. Second-class postage paid at Redwood City and at additional entry points.

Article Submission: Please write the editors at Micro/Systems Journal, Box 1192, Mountainside, NJ 07092 or call (201) 522-9347 to see if there is interest in a particular subject. Please do not send the article unless we ask for it.

If you are interested in reviewing hardware or software please write telling us your interests, your background, and include a sample of your writing.

Send a stamped self-addressed business size envelope (to NJ address) for a copy of our Author's Guide.

Address Correction requested: POSTMASTER: send address changes to Micro/Systems Journal, 501 Galveston Dr., Redwood City, CA 94063. **ISSN 8750-9482**

Change of Address: Please send old label and new address to: Micro/Systems Journal, 501 Galveston Dr., Redwood City, CA 94063.

Correspondence: Editorial Office: Box 1192, Mountainside, NJ 07092; (201) 522-9347. The editor may also be contacted via MCI Mail (SLIBES). Business Office: 501 Galveston Dr., Redwood City, CA 94063; (415) 366-3600.

Advertising Rates: Available on request. Call (415) 366-3600 or write to: Micro/Systems Journal, 501 Galveston Dr., Redwood City, CA 94063.

Subscription rates: U.S.: $20 for one year, $35 for two years. Canada & Mexico add $7 per year for airmail; $3 per year surface mail. Other countries add $12 per year for airmail.

Entire contents copyright © 1987 by M & T Publishing, Inc.; unless otherwise noted on specific articles. All rights reserved.

Editor's Page

Sol & Don Libes

DOS Is Getting to Look a Lot Like UNIX

If you have even the slightest familiarity with UNIX, you have probably noticed the amazing resemblance to UNIX that DOS is taking on as it matures. Indeed, in its current state, DOS might almost be considered a single-user version of UNIX. Let's take a look at the similarities.

THE FILE SYSTEM

Beginning with Version 2.0, DOS provided a UNIX-like hierarchical file system with nested directories. Just like UNIX, the command processor offers a *path* variable, and the commands for moving through the file hierarchy are exactly the same, including the UNIX abbreviations for current directory (.) and parent directory (..).

One tiny difference is that UNIX uses slash (/) to separate directories whereas DOS uses backslash (\). I think the developers did this on purpose to state that "the UNIX file system is great, but admitting that we are copying it would hurt too much." Or maybe they were worried about some copyright. In any case, except for that slash, UNIX natives can feel quite at home on DOS file systems.

Although the internal structure of UNIX file systems (inodes) is different from DOS' (FATs and raw disk sectors), there are enough similarities that it is easy to recreate most of the UNIX file utilities for DOS with little change in functionality. The little that is missing relates to multiuser support, such as file protection.

COMMAND PROCESSING

A notable UNIX innovation allows users to choose their own command processor, possibly home-brewed. DOS provides the same capability with COMMAND.COM. You can replace it with your own command interpreter.

Several different replacements for COMMAND.COM are available in the public domain and as commercial products. Numerous extensions also exist to provide graphics and multitasking help. There are also versions of the popular UNIX shells, allowing you to have a UNIX shell as your DOS command processor.

Typical UNIX shells are complete languages, featuring variables and control constructs such as do-whiles and if-then-elses. These features allow users to join programs and "tools" to build more complex programs quickly. Although not efficient in speed of execution, they provide extremely fast prototyping, which is often more critical than final run-time speed.

These *shell scripts* are almost completely duplicated by the DOS BAT feature, which allows sequences of commands to be executed automatically.

FILTERS, PIPES, AND REDIRECTION

Additional UNIX features incorporated in DOS are those of redirection, pipes, and standard input and output.

Redirection lets users change the standard input and output from the keyboard and screen to any other file or device. DOS implements the most important features of these and uses the very same characters as UNIX does (> for output, < for redirect input, and | for pipe).

Using pipes you can take programs that produce output on the screen and redirect them to programs that take input from the the keyboard. Programs designed to read and write in this way are called filters because they take their input, perform some transformation, and provide it to the next program for processing.

Filters are among the best kinds of programs to have for tool building because they can be plugged into place when needed. It is typically easier to write a filter style of program because you make no assumptions about the devices or files your data is coming from or going to.

Three filters are provided with DOS (*sort*, *find*, and *more*) and many others are available in the public domain and as commercial programs. Some of these DOS filters have different names from their UNIX counterparts, but most are the same.

I/O DEVICE INDEPENDENCE

Device independence allows you to take a program that writes on a terminal and redirect it to write on a disk, a tape, or whatever. Although there are obvious limitations (for example, you can't store files on a terminal, you can't change the baud rate of a tape drive), UNIX carries this idea off rather well. All device drivers have a common, well-defined set of entry points.

DOS is coming along but is not quite there yet. Like CP/M, the BIOS defines the basic I/O services available to the programmer. The BIOS then makes calls on device drivers external to DOS and supplied by the user (or manufacturer of the device). Unfortunately, the BIOS is not standardized and can be modified by resellers. This explains why programs written for PC-DOS/MS-DOS won't run on XYZ-DOS.

Once a BIOS is standardized, however, programs and device drivers do tend to be portable. Hence a program written for one PC-DOS/MS-DOS machine (as an example) is very likely to run on another manufacturer's PC-DOS/MS-DOS machine.

CONCLUSION

Recent announcements by Microsoft indicate that the new DOS will share further similarities with UNIX. It will:

1. use protected mode—the system cannot be corrupted by the user
2. remove memory address restriction and support virtual memory
3. allow multiple processes with multitasking in each process
4. offer dynamic linking and shared libraries—resulting in much more efficient use of memory and greater multiprocessing flexibility
5. allow interprocess communication—necessary for concurrent processes to communicate with each other

Although none of these attributes are what make UNIX great (they are common to many large operating systems), having these features in DOS will make it closer than ever to UNIX. It is remarkable how much of the power of UNIX has been incorporated into DOS. §

TURBO PROFESSIONAL™

SERVICE INTERRUPTS
No assembly required

RESIDENT PROGRAMS
Easy, pop-up routines

EXECUTIVE PROGRAMS
Run ANY DOS program

DISK SECTOR I/O
Lowest level access

FAST TEXT WINDOWS
Virtual windowing system

KEYBOARD MACROS
Simple, powerful

LOTS OF EXAMPLES
21+ full example programs

MUCH MORE . . .
Over 140 routines in all

"If you never thought Turbo Pascal was a systems programming language, you've never seen Turbo Professional."
Darryl Rubin
Computer Language

For programs that move with technology—Turbo Professional—a truly professional library of subroutines.

150 page reference manual. Full source—many example programs.
No royalties charged for applications.

Requires IBM compatible, DOS version 2.0 or greater, Turbo Pascal 2.0 or greater.

Sunnyhill Software not affiliated with Borland International.
Turbo Professional, trademark of Sunnyhill Software
Turbo Pascal, registered trademark of Borland International

Dealer Inquiries Invited.

$69.95 plus $5.00 S&H

Washington residents add 7.9%
International orders add $5.00
VISA and Mastercard accepted.

To order Toll-Free
call 1-800-367-0651

Sunny Hill Software
P.O. Box 55278
Seattle, WA 98155
(206) 367-0650 M-F, 8-6 PDT

TASKVIEW™

EASY TO USE

OPTIONAL MENUS

TIME SLICING

VIRTUAL MEMORY

EMS SUPPORT

EGA SUPPORT

PRIORITY DRIVEN

CUT AND PASTE

Compatible, efficient DOS multi-tasking.

We designed Taskview with efficiency in mind. During normal operation, TASKVIEW hides behind DOS, providing you with control of up to 10 concurrent or non-concurrent programs. Just the touch of a key instantly switches a program to the foreground. Included desktop utilities let you cut and paste from program to program. Simple to use and reasonably priced, no well equipped PC user should be without it.

Requires: PC/AT/Jr compatible, DOS 2.0-3.1, 256K RAM, 1 Floppy drive.

Sunnyhill Software not affiliated with Borland International
Taskview trademark of Sunnyhill Software
Sidekick registered trademark of Borland International

Dealer Inquiries Invited.

$79.95 plus $3.00 S&H

Washington residents add 7.9%
International orders add $5.00
VISA and Mastercard accepted.

To order Toll-Free
call 1-800-367-0651

Sunny Hill Software
P.O. Box 55278
Seattle, WA 98155
(206) 367-0650 M-F, 8-6 PDT

UNIX TOOLS FOR YOUR PC

PC/VI™

UNIX's VI Editor Now Available For Your PC!

Are you being as productive as you can be with your computer? An editor should be a tool, not an obstacle to getting the job done. Increase your productivity today by choosing **PC/VI**—a COMPLETE implementation of UNIX* VI version 3.9 (as provided with System V Release 2).

PC/VI is an implementation of the most powerful and most widely used full-screen editor available under the UNIX operating system. The following is only a hint of the power behind **PC/VI:**

- Global search or search and replace using regular expressions
- Full undo capability
- Deletions, changes and cursor positioning on character, word, line, sentence, paragraph, section or global basis
- Editing of files larger than available memory
- Shell escapes to DOS
- Copying and moving text
- Macros and Word abbreviations
- Auto-indent and Showmatch
- MUCH, MUCH MORE!

Don't take it from us. Here's what some of our customers say: "Just what I was looking for!", "It's great!", "Just like the real VI!". "The documentation is so good I have already learned things about VI that I never knew before." — *IEEE Software*, September 1986.

PC/VI is available for IBM-PC's and generic MS-DOS‡ systems for only $149. Included are CTAGS and SPLIT utilities, TERMCAP function library, and an IBM-PC specific version which enhances performance by as much as TEN FOLD!

PC/TOOLS™

What makes UNIX so powerful? Sleek, Fast, and **POWERFUL** utilities! UNIX gives the user not dozens, but hundreds of tools. Now the most powerful and popular of these are available for your PC! Each is a complete implementation of the UNIX program. Open up our toolbox and find:

• BANNER	• DIFFH	• PASTE	• SPLIT
• BFS	• DIFF3	• PR	• STRINGS
• CAL	• GREP	• RM	• TAIL
• CHMOD	• HEAD	• SED	• TR
• CUT	• MAKE	• SEE	• TOUCH
• DIFF	• OD	• SORT	• WC

All of these for only $49.00; naturally, extensive documentation is included!

PC/SPELL™

Why settle for a spelling checker which can only compare words against its limited dictionary database when **PC/SPELL** is now available? **PC/SPELL** is a complete implementation of the UNIX spelling checker, renowned for its understanding of the rules of English! **PC/SPELL** determines if a word is correctly spelled by not only checking its database, but also by testing such transformations as pluralization and the addition and deletion of prefixes and suffixes. For only $49.00, **PC/SPELL** is the first and last spelling checker you will ever need!

Buy **PC/VI** and **PC/TOOLS** *now* and get **PC/SPELL** for only $1.00! Site licenses are available. Dealer inquiries invited. MA residents add 5% sales tax. AMEX, MC and Visa accepted without surcharge. Thirty day money back guarantee if not satisfied! Available in 5¼", 3½" and 8" disk formats. For more information call today!

*UNIX is a trademark of AT&T. ‡MS-DOS is a trademark of Microsoft.

CUSTOM SOFTWARE SYSTEMS
P.O. BOX 678 • NATICK, MA 01760
617•653•2555

CSS

UNIX TOOLS FOR YOUR PC

News, Views & Gossip

by Sol Libes

RUMORS & GOSSIP

Intel is rumored to be developing a new version of the 80386 that is pin-compatible with the 80286. In other words, it has 16-bit I/O but otherwise is a true 32-bit processor (à la the 8088/8086). Imagine unplugging the 80286 from your AT and replacing it with an 80386 and running 386-DOS and Windows-386.

Intel is also rumored to be about to produce its first silicon versions of the 80486 microprocessor and has begun defining the architecture of an 80586. The 486 is expected to be an enhanced version of the 386, and speculation is that the 586 will be designed for parallel processing systems.

Prices of 2,400-baud Hayes-compatible modems are expected to drop sharply to well under $200 this fall as chip manufacturers integrate functions in LSI chips.

Ashton-Tate is expected to release its new version of dBASE in November at Comdex. Look for greatly increased speed, SQL support, pull-down menus, a new file system, and expanded report capability.

UNIX appears to be gaining in popularity. The three largest Japanese computer makers (**NEC**, **Fujitsu**, and **Hitachi**) have announced new medium-size machines using UNIX. And **Microsoft** and **AT&T** have agreed to jointly develop and market an 80386 version of UNIX.

Dataquest, a respected market-research concern, is predicting that sales of personal computers will increase 8 percent this year compared to 1986. Last year saw a comparable increase. These moderate increases are far below the double-digit increases of prior years. Because the cost of systems is decreasing, this means that about 15 percent more systems will be shipped. New systems from Apple and IBM as well as lower prices for PC clones are expected to fuel the increase.

Japanese companies are moving into large-quantity production of 1-Mb RAM chips. Thus, you can expect all new systems to have memory systems of 1-MB and more. The 640K memory limit of the PC will soon be a thing of the past. These companies are also beginning to sample CMOS 1-Mb RAM chips and expect to be producing them by the end of the year. Thus, next year's portable systems will also sport megabyte memories.

In the meantime, papers were presented at the International Solid State Conference in February describing the fabrication of 4-Mb and 16-Mb RAM chips (imagine a 2-MB RAM in a single chip). Papers were also presented on memory devices having 1-ns access times and microprocessors with 30-MHz clock rates.

286-DOS & IBM

The new version of PC-DOS for 286-based machines is now in the final stages of development, with formal announcement expected this summer and shipment in the late fall. **Microsoft** is calling it NewDOS; no word yet on what IBM will call its version. It will sport a new graphics/icon/mouse-oriented user interface (à la the Macintosh). Gone will be the old DOS command-line user interface.

Users who bought IBM PC/AT machines or compatibles, expecting to run the new operating system, are going to be quite put out. It will not run on current AT-type systems. Although the new operating system has been created by Microsoft, IBM has customized it to run only on a new 286-based system to be released at the same time as the new DOS.

IBM has used this marketing tactic in the mainframe area a number of times—releasing hardware before the system's software is ready and forcing the customer to run old software that does not take advantage of the hardware. In this case purchasers of AT systems have been forced to run either PC-DOS 2.x or 3.x, the exact same operating system that runs on the PC and PC/XT. All users have gotten faster operation, while most of the 286's facilities have gone unused and performance has been far below the hardware's capability.

Now AT system owners will be forced to dump their gear and buy the new IBM 286-based system to obtain an operating system that takes full advantage of the 286's features. Not only that, they will have to upgrade or change software to get true 286 performance.

Microsoft's version of the operating system will lack the custom features added by IBM but should contain Windows hooks. It appears that IBM has exerted pressure on Microsoft to hold up release of its OS until IBM has released its version.

OEMs that adopt NewDOS will no doubt make an all-out attempt to make the system compatible with the IBM system. This should take them at least 6 months, and more likely much longer, because they will have to design both hardware and software. IBM is expected to integrate its display, disk, I/O, mouse, and network controllers onto the mainboard, so it will take the cloners a long time to develop compatible systems. And, by then IBM will have made some changes to keep the competition off balance. This is the strategy that has worked so successfully for IBM in the mainframe area—it well may work again in the desktop arena.

IBM, however, faces problems here that may cause users to not go for this new system. First, application software for the system will not be readily available. Second, 386-based systems, providing performance far better than 286-based systems, are already available. The **Compaq-386** will have been out almost a year by the time IBM introduces its new 286 system. Further, 386 operating systems are already available that provide high-performance multitasking and multiuser facilities and Microsoft is close to releasing Windows-386. The Compaq-386 is already well established in the marketplace, and it is anticipated that Compaq will have more than 100,000 386 systems in place by the time IBM releases its new 286 system.

386 SYSTEM UPDATE

The 80386 scene is quickly developing into the chaotic marketplace that everyone predicted it would without a dominant player such as IBM. Despite numerous announcements of 386 desktop systems, **Compaq Computer** (Houston, Texas), **Advanced Logic Research** (Irvine, California), and **Kaypro** (Solana, California) are the only companies currently shipping 386 systems in quantity. Prices for a basic system with hard disk, EGA display system, and 1-MB of RAM range from $7,000 to $8,000. PC's Limited and several other mail-order manufacturers are expected to start shipping 386 systems shortly. And, **Intel** has indicated that it expects to produce a half million 80386 chips this year, with probably half going into desktop systems.

Multitech Electronics (Sunnyvale, California) has announced a 386 system that is compatible with the Compaq Deskpro 386

Uninterruptible Power Systems

Read what the experts say;

"Clary Corp's OnGuard 600VA has the makings of the perfect UPS: it's functionally invisible. Whether or not line voltage is available, it constantly supplies up to 600VA of nearly pure sine-wave power at an almost unvarying 120 volts. No UPS is more unobtrusive than the OnGuard."

-Winn L. Rosch,
PC Magazine,
SEPTEMBER 16, 1986

For heavy-duty use with large systems, our choice is Clary Corp.'s OnGuard 600VA. It has sufficient capacity to run a network server with external hard disk(s) long enough to bridge most outages or shut down the network in an orderly manner. Also, it's quiet enough for an office area and simple enough to use that it can be forgotten.

-PC Magazine,
Editor's Choice,
SEPTEMBER 16, 1986

PROTECT YOUR IBM SYSTEM/34/36
FOR ONLY

$4590.00*

AVAILABLE IN 600VA, 800VA, 1.0 KVA, 2.0 KVA & 2.5 KVA

- Capable of providing nearly twice its rated output capacity (with switching power supply loads).
- Designed specifically for use with your IBM system/34/36, PC, XT and AT.
- On-line operation, static bypass switch and sinewave output.
- Fifth generation, high frequency PWM technology reducing size and weight by over 50%.
- Ideal for multiple XT and AT configurations
- Meets FCC Part 15-Class A, NEC and IEEE 587 standards.
- Virtually silent.

CLARY®

ONGUARD™

CLARY CORPORATION
320 West Clary Avenue
San Gabriel, CA 91776 (818) 287-6111 x645

***2.0 KVA IN QUANTITIES OF 4.**

at a price about 40 percent lower. This may actually help Compaq by, in effect, making the system a standard. If one or two more companies introduce systems to this "hardware standard" and **Microsoft** releases a 386 operating system before IBM releases its 386 system, IBM may have serious competition in this market.

Compaq is expected to announce a portable 386 system shortly. And, Intel is shipping its 386 system board to OEMs. The board has an AT form factor and some clone makers have already introduced systems using this board. The board alone can be purchased for $3,000 from some mail-order outlets. I expect this board to be copied by some Far-East clone component manufacturers and for prices to drop to half before the year is out. One mail-order firm is already using the board and is advertising a 386 system with 1.2-MB floppy, 20-MB hard disk, and monochrome display for $3,350 (**Lolir**, 13933 N. Central, #212, Dallas, TX 75243; (214) 234-8056). And **Computer Classifieds** (17830 Rd. 9, Miami, FL 33162; (305) 651-5853) is advertising a $3,195 system that includes 640K RAM, 1.2-MB floppy, and I/O.

Chips and Technologies, the IC manufacturer that brought out chip-sets for the EGA and AT motherboard and enabled cloners to undercut IBM's prices, has introduced a chip-set for the 80386. This is expected to cause a substantial drop in the prices of 386 systems. The first company expected to bring out a 386 board using this chip-set is **Zeos International** (530 5th Ave. NW, St. Paul, MN 55112; (612) 633-4591); look for a base price of less than $2,000. I expect the 386 motherboard price to drop to less than $1,000 by this time next year (consider that AT motherboards already sell for well under $500). The Zeos board uses 12 C&T chips and the Phoenix BIOS and can accommodate up to 16-MB of zero-wait-state RAM on-board.

Intel has also announced that it is shipping samples of a 24-MHz 80386 and new support chips that will bring 386 systems into the true minicomputer class of operation.

In the meantime, none of the big-name software manufacturers have indicated their intention to release any software specifically designed for 386 systems this year. Current 386 system purchasers are running standard DOS software at two to three times the processing speed of a standard AT system. Users of high-speed AT systems are finding that the difference in throughput is not significantly better between a 16-MHz 386 and a 12-MHz (zero wait state) AT clone. The 386 systems are most popular for CAD, desktop publishing, and network file server applications.

XENIX-386 has been announced by **Santa Cruz/Microsoft/Interactive Systems** and should be available this fall. It will have Xwindows and Ethernet interfaces. Compaq has already committed to carrying it.

Locus Computing is shipping the Merge-386 integrated DOS/UNIX operating system ($400), which allows concurrent execution of DOS and UNIX applications. Locus claims that Intel, Convergent Technologies, TeleVideo, and Microport will carry it.

Digital Research (Concurrent DOS-386), **Softguard** (VM-386), and **Software Link** (PC-MOS/386) have all announced 386 multitasking, multiuser DOS-compatible operating systems. All are in beta test and are expected out shortly.

CROMEMCO SOLD

Cromemco (Mountain View, California), one of the pioneers in S-100 components and systems, has been acquired by **Dynatech Corp.** (Burlington, Massachusetts). Cromemco was founded in 1974 and run by Harry Garland and Roger Melen as a sideline business to make computer kits. Harry and Roger were electrical engineering teachers at Stanford University. When the Altair was introduced in 1975, they began designing S-100 plug-in cards for the unit. They gained a reputation for innovative designs and very high-quality products.

In 1976, they quit teaching and were the first to introduce a Z80-based CPU card and complete system. They later introduced a 68000-based system and high-performance video-display systems. In 1980 they introduced a small Z80-based desktop system, which proved to be very successful, particularly outside the U.S. The company reached its peak in 1984, with sales of $40 million and 400 employees. However, sales from then on dropped down rapidly to $15 million in 1986, forcing a cut back to 100 employees and a consolidation from two buildings to one. Dynatech reportedly paid $5 million for the company. Harry and Roger will continue to run it.

The company has concentrated on providing systems for video graphics and data communications. It is expected to introduce a system based on the Motorola 68030 soon.

WINDOWING STANDARD FOR UNIX ADOPTED

Eleven companies have announced support for the X Window System, a graphical windowing standard for UNIX systems that is hardware and operating system independent. The companies are **Adobe Systems, Apollo Computer, Applix, Dana Computers, Data General, Digital Equipment Corp., Hewlett-Packard, Massachusetts Computer, Siemens, Sony,** and **Stellar Computer**. Significantly missing from the group are **Sun Microsystems, IBM,** and **AT&T** (I suspect that they will sell versions to their customers).

The object is to add a Macintosh-style "user friendly" interface to what is considered the most "user hostile" operating system. The X Window System was developed at MIT and consists of a set of graphical windowing primitives written in C. Several of the above companies already have X Window running on some of their systems.

At a recent Usenix conference, a networking system was installed in the exhibit hall and virtually all the workstation vendors were tied into it and running X Windows with demo programs being freely transferred from system to system. There is no doubt that this kind of standardization will encourage application software developers to develop software for X Windows.

IEEE-696 (S-100) STANDARD NEWS

Richard Kalish has taken over the chairmanship of the IEEE IEEE-696 (S-100) Standard Committee. He is the Director of Hardware Development at **CompuPro/Viasyn** (26538 Danti Ct., Hayward, CA 94545). Under his direction, a vote has been taken on a change to the standard to allow regulated power supplies to be used with boards that do not contain regulator ICs. It looks as though the final vote will be positive and that this change will be made to the standard. It will significantly reduce production costs of S-100 systems, making them more competitive.

Also Richard is forming a study group to work on a 32-bit revision to the standard. Readers interested in participating in this work, or in other efforts related to the S-100 standard, should contact him.

I expect that several manufacturers will soon introduce 32-bit S-100 board products. These are expected to be single-board computer designs (integrated CPU, RAM, and ROM) and hence will comply with the standard. We expect to review these products as they are released. §

COMPETITIVE EDGE

1421 E. Goldsmith • Plymouth, MI 48170 • **(313) 451-0665**

SPEED SHOP

SINCE 1980

100% AT ™ COMPATIBLES 286's & 386's

Velocity 286-12™ 12 MHZ, 1024K, 40MB, 28ms Hard Disk, EGA Card P/S/C	**$2795.**
Velocity 286-10™ 10 MHZ, 1024K, 40MB, 28ms Hard Disk, EGA Card P/S/C	**$2295.**
Velocity 386-16™ 16 MHZ, TRW Service, 40MB, 28ms, H.D., EGA Card P/S/C, 1024K	**$4095.**
Velocity 386-20™ 20 MHZ, Many Options, 40MB, 28ms Hard Disk, EGA Card P/S/C	**SOON**

100% XT COMPATIBLES

Turbo XT™ 4.77/8 MHZ, 640K, 20MB Hard Disk, 360K, Floppy-Mono & Monitor	**$1195.**
Turbo XT™ 4.77/8 MHZ, 640K, 30MB Hard Disk, 360K, Mono Card & Monitor	**$1295.**
Turbo XT™ 4.77/8 MHZ, 640K, 20MB Hard Disk, 360K, Std. Color Card & Monitor	**$1369.**
Turbo XT™ 4.77/8 MHZ, 640K, 30MB Hard Disk, 360K, Std. Color Card & Monitor	**$1469.**
Turbo XT™ 4.77/8 MHZ, 640K, 20MB Hard Disk, 360K, EGA Card & Monitor	**$1695.**
Turbo XT™ 4.77/8 MHZ, 640K, 30MB Hard Disk, 360K, EGA Card & Monitor	**$1795.**

Hard Disks Available to 224 Megabytes on XT or AT™ Compatibles

HARD DISK DRIVES

ST4038, 30 MB, 39ms	**$595.**
ST4051, 40 MB, 39ms	**$699.**
Miniscribe 6053 42MB 28ms	**$699.**
ST225 20MB, 65ms	**$310.**
Miniscribe 70MB, 28ms	**$995.**
Toshiba 70MB, 25ms	**$1095.**

MONITORS

NEC Multi Sync EGA	**$569.**
Teco EGA 640x350	**$425.**
Teco Std. Color	**$309.**
Samsung Amber Flat	**$119.**
EGA Card, EGA, CGA, MPA	**$209.**
Everex EGA	**$229.**

BASE AT $995.

1.2 MB Floppy-Hard Disk/Floppy Controller, 512K RAM, AT™ Style Keyboard Clock, 192 Watt Power Supply, Set-Up Software, 1 Year Limited Warranty.

NOVELL, XENIX™, ALLOY SOLD & SUPPORTED

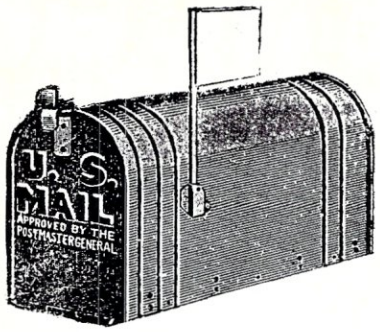
12 MHZ 80286 Zenix System, 8 Serial, 1024K, 36MB, 28MS	**$4095.**
3 User Alloy Slave System, 8MHZ, 80286, 30MB, H.D., 1024K	**$4195.**
2 User (2) AT Compatibles, 640K, 8MHZ, 80286's, (1) 30MB, H.D. Novell 86	**$5295.**

2, 4, 8, 16, 31 USER SYSTEMS AVAILABLE

S-100, All CompuPro®, Lomas, Teletek Boards at Lowest Prices.

Desktop Publishing Systems and upgrades available.

XT & AT are trademarks of International Business Machines. CompuPro is a registered trademark of VIASYN. Xenix is a trademark of MicroSoft.



there is mail . . .

We welcome your letters, with their comments, compliments, criticism, and suggestions. We do not have the staff to answer all letters personally. All letters become the property of M/SJ *and may be subject to editing. We do not print letters that do not include a name and address.*

Please send your letters to: Micro/Systems Journal, *Box 1192, Mountainside, NJ 07092.*

S-100 STILL POPULAR

Dear *MS/J*:

It's reassuring that *Micro/Systems Journal* is now being published by the same people who publish *Dr. Dobb's Journal*, of which I am a loyal reader. The Editor's Page and News, Views & Gossip are my favorite reading in *M/SJ*.

I wish you the best and hope that you will maintain the same editorial continuity. The different S-100 bus products are, for me, very important.

Z. Kaminski
Opglabbeek, Belgium

Thanks for the pat on the back. Many other readers have also told me they really like my columns. I certainly plan to continue my ramblings and gossiping.

The S-100 bus market is now very mature and no longer plays the dominant role it once played in the U.S. microcomputer marketplace. It is now confined almost exclusively to the multiuser/networking PC-compatible system market. The number of S-100 product manufacturers is now a fraction of what it was only five or six years ago. The S-100 product market is still viable, though, and we expect to continue to support it in M/SJ.

—Sol

LOOKING FOR LOW-COST AT CLONE

Dear *MS/J*:

Congratulations on finding a fine publisher. I enjoy reading your journal but think it is getting a little heavy on articles aimed at the IBM PC and compatibles. The occasional articles on good old CP/M-80 and the Z-80 are very welcome. I would enjoy seeing more articles on the 68000, such as the one on bringing up CP/M-68K that appeared in the November/December 1985 issue. You should publish a yearly index.

As yet I do not own an IBM PC or compatible, so I was intrigued by the opening story in your News, Views & Gossip column in the November/December 1986 issue. While in San Jose on business, I tried to contact HiTech International, but the directory-assistance operator had no listing for it. Could you give me the company's address and phone number, please?

Gary Kaatz
Hoffman Estates, IL

I finally got around to producing an index for 1985 thru 1986. It appeared in the January/February 1986 issue. I hope to produce it on an annual basis from here on.

Although many of our readers continue to use CP/M-80, there is really nothing new in this area. There are no new software products, and Digital Research does not plan to upgrade it and in fact no longer even supports it. Rather, we see many of our readers who run Z-80-based systems turning to the ZCPR operating system, which is CP/M-80 compatible. We have run several articles on ZCPR during the last two years and will publish more. The Hitachi 64180 Z-80-compatible microprocessor offers significant improvement for 8-bit users. We ran an article last year on a low-cost way of converting a Z-80 system to a 64180 system. And Echelon (it advertises in M/SJ*) has tailored its operating system to take advantage of the chip.*

HiTech International is located in Milpitas, California, and its telephone number is (408) 263-3300. Its AT-compatible system costs $995, runs at 6/8 MHz, and includes a 1.2MB floppy, a combo disk controller, two serial ports, and an RGB video controller board with parallel port. It is certainly the lowest price I have seen yet for an AT clone.

Several other companies are also selling AT clones for less than $1,000 or very close to it. I suggest you check my article on the subject of assembling AT clones in this issue. —Ed.

SEARCHING FOR A CP/M GRAPHICS STANDARD AND Z80 MEMORY-MANAGEMENT INFO

Dear *MS/J:*

Like most other readers I would prefer less IBM PC material, although some is valuable.

What I would like to see is a graphics standard of some sort that would allow CP/M users to build graphics extensions to their existing systems and an evaluation of Z80 memory-management systems (not bank-selected), which I believe some manufacturers have developed on a page-mapped philosophy.

I hope you have continued success in the venture, as I believe it's very important to us CP/M and S-100 hackers.

Peter Trigger
Brisbane, Australia

There is no doubt that a large number of our readers are Z80, CP/M, and S-100 hackers. We will continue to provide support for these readers. We have determined, however, that most of these readers are making a transistion from 8-bit systems to 16-bit ones. In most cases, this means PC, PC/XT, and PC/AT-compatible systems. I made this transition five years ago, in fact.

Digital Research Inc. made an effort to establish a graphics software interface standard for CP/M-80 many years ago. It was called GSX. We provided details on it in the old Microsystems *magazine. Regretfully, it was introduced just as the PC, with its graphics standard, was introduced and never achieved any acceptance.*

As to Z80 memory management, I do not know the distinction between bank-select and page-mapped. I would appreciate hearing from readers who have implemented Z80 memory-management systems.

Finally, keep in mind that virtually all S-100 manufacturers today are selling multiuser, 16-bit, CPU-based, MS-DOS-compatible systems. This is really the only area in which S-100 manufacturers can compete successfully. We support this effort and have run many articles to address this area.—Ed.

INSTANT EXPERTISE!

MASTER*KEY

BEGINNERS FIND IT IRRESISTIBLE! EXPERTS FIND IT INDISPENSABLE!

THE FOLLOWING COMPANIES OR INDIVIDUALS WORKING FOR THESE COMPANIES HAVE USED MASTER*KEY

AMERICAN NUCLEAR INSURERS • AB DICK • ADVANCED GENETIC SCIENCES • ALCOA • AMERICAN N
ATOMIC ENERGY OF CANADA LIMITED • AUBURN UNIVERSITY • AVCO SY
BAUSCH & LOMB • BECKMAN INSTRUMENTS • BIONETICS RESEARCH INCOR
BORGESS MEDICAL CENTER • BROOKHAVEN NATIONAL LABORAT
CALIFORNIA TEACHERS ASSOCIATION • CHAMPION SPARK PLUG
CLEVELAND STATE UNIVERSITY • CLINICAL UROLOGY INCORP
IDAHO • COMMONWEALTH PUBLICATIONS • COMTEL • C
PER BIO MEDICAL • COOPERS & LYBRAND • CRAY
UNITY COLLEGE • DETROIT EDISON • DIGITA
OUGLAS AIRCRAFT COMPANY • DOW CORN
YNASOFT • EAGLE ELECTRONICS • EA
EDERAL JUDICIAL CENTER • FIDEL
GENERAL ELECTRIC • GEO
GRUMMAN AEROSPA
OR CORP • INTER
ICAL CENTER
ATIONAL

***instantly and automatically with a turn of* MASTER*KEY**

MASTER*KEY - No other product comes close!

An EXPERT may not know the answer, but always knows where to find it.

MASTER*KEY helps experts AND beginners find solutions to nagging, confusing, and frustrating software problems more quickly and easily than any other software available, at any cost! It gives you know-how within hours that may otherwise take years of experience.

MASTER*KEY - Improved, unbelievably easy to use!

The original MASTER*KEY has enjoyed phenomenal success over the past year. With response from previous users and comparisons to similar products, it has been completely rewritten to be more comprehensive, faster, and even easier to use. It can split the display screen to view two different disassembled areas of a program at the same time.

MASTER*KEY works both automatically from the DOS command line or interactively from menus similar to Lotus Corporation's 1-2-3 or Symphony, no need to remember any new commands or continually refer to a manual. Use it immediately!

MASTER*KEY - Smart, educational!

MASTER*KEY is an intelligent MS-DOS Reverse Assembler. It rapidly wades through massive and baffling object code files to effortlessly find potential trouble spots. It identifies all branch addresses, ROM BIOS and DOS functions and interrupts, data, and code to change or remove. All references are cross-referenced by MEMORY LOCATION!

MASTER*KEY will take any program from your IBM-compatible computer and return fully-documented assembly language source code. Since information produced is documented, YOU DON'T NEED TO BE AN ASSEMBLY LANGUAGE PROGRAMMER. If you desire, it will help you learn assembly language faster and better since you work with any existing program. It can produce a Microsoft MASM-compatible assembly language source code file that may be edited and reassembled to create a new program from any executable MS-DOS file.

Minimum system requirements:

256K + 8088/8086/80186/80286/80386 PC
MS-DOS or PC-DOS 2.0-3.2
One 360K DSDD Floppy Drive
(IBM PC Format)

MASTER*KEY should not be confused with any public domain or share ware software that may have a similar name or be a similar product.

turn this

```
A>DEBUG PROGRAM.COM
-D100 12B
96F1:0100  EB 14 50 52 4F 47 52 41-4D 20 43 41 4E 4E 4F 54  k.PROGRAM CANNOT
96F1:0110  20 52 55 4E 0D 0A 50 B4-30 CD 21 86 E0 3D 36 01   RUN..P4OM!.`=6.
96F1:0120  73 09 BA 06 01 B4 09 CD-21 CD 20 58              s.:..4.M!M X
```

NOTE: The output is totally microsoft MASM compatible.
NOTE: The cross-reference is by memory location within the program file!

into this

```
H00100: JMP     Short H00116                    ;00100 EB14          __
;-------------------------------------------------
        DB      "PROGRAM CANNOT RUN"            ;00102
        DB      0Dh                             ;00114
        DB      0Ah                             ;00115
H00116: PUSH    AX                              ;00116 50            P
        MOV     AH,30h                          ;00117 B430          _0
        INT     21h             ;1-DOS_Ver_Number ;00119 CD21        _!
        XCHG    AH,AL                           ;0011B 86E0          __
        CMP     AX,0136h                        ;0011D 3D3601        =6_
        JNB     H0012B          ;. . . . . . . . ;00120 7309        s_
        MOV     DX,0106h                        ;00122 BA0601        ___
        MOV     AH,09h                          ;00125 B409          __
        INT     21h             ;1-Display_String ;00127 CD21        _!
        INT     20h             ;TERM_normally  ;00129 CD20          _
;-------------------------------------------------
H0012B: POP     AX                              ;0012B 58            X

        0106h                   :   122
        0136h                   :   11D
        1-DOS_Ver_Number        :   119
        1-Display_String        :   127
        H00100                  :   100
        H00116                  :   100   116
        H0012B                  :   120   12B
        TERM_normally           :   129
```

Phone Orders Accepted on MC or VISA. Please call **(714) 596-0070**

Mail to: Sharpe Systems Corporation 2320 E St., La Verne, CA 91750

YES! I want to unlock my programs. Send **MASTER*KEY.**

☐ I have enclosed a check for $79.95.

☐ Please send my order C.O.D. (Cashier's check or M.O. only) add $2.50 shipping.

California residents add 6.5% sales tax.

Name _____

Company Name _____

Address _____

City _____ State _____ Zip _____

$79.95

Not copy protected

Sharpe Systems Corporation
2320 E STREET, LA VERNE, CA 91750 714-596-0070

MS-DOS is a trademark of Microsoft. PC-DOS is a trademark of IBM.

Dealer/Distributor Inquiries Welcome

The C Forum

Don Libes

For a Good Time, Call . . .

This column features tips and techniques for using the C language productively. It discusses typical problems with using C and their solutions. Reader suggestions, comments, and questions are encouraged. Address them to "The C Forum," Micro/Systems Journal, *Box 1192, Mountainside, NJ 07092.*

This month, I present a short but useful application to set your computer's clock to a national reference—the Naval Observatory in Washington, D.C.

Having a precisely set clock may not seem important, but it can be. It is often critical when systems are interconnected. Suppose, for example, you are using *make* to compile programs, and some files are stored on your local disk and others are stored on a networked disk. If the network file server is set for a later time than is your system, it is possible for someone to modify files on the network file server. *Make* would then fail because the new file looks older than it really is.

Another example is when several systems are being backed up across a network. If the remote systems dates are newer than the system that is controlling the backup, modified files will not get backed up. These problems and others like them can be extremely difficult to track down—better to prevent them from occurring in the first place. As systems are interconnected, working from the same date will be just as important as working with the same character set.

Most systems have a real-time clock onboard. Implemented in hardware, it is very reliable. To the application programmer, it is useful in only one way—reading the current time of day.

When the clock is incorrect, however,—say, if the battery has run down or when it is first powered up—the clock can also be set.

What the program in Listing 1 does is simple. It calls up the Naval Observatory in Washington, D.C., which keeps time for the military and provides time stamps in ASCII at 1,200 baud to whomever dials it up. After connecting to this system, the program requests the time, and upon receiving it, it sets the time locally.

Here is the main program to do this:

```
main(argc,argv)
int argc;
char **argv;
{
options(argc,argv);
select_modem();
setup_modem();
dialup();
login();
gettime();
settime();
logout();
cleanup();
}
```

This program has several dependencies based on your operating system, computer, and modem that should be easy to configure.

I will go through each procedure in the order called from *main( )*, discussing the program and mentioning the particular dependencies, if any.

options(argc,argv);

These are the process command-line options. There are two important options. If the user specifies *-l*, the following field is taken as the name of the modem to be used (for example, COM1:). Specifying *-s* makes the program actually set the system time. Without it, the program goes through the motions of dialing up the Naval Observatory and retrieving the time but does not do anything with it.

The remaining options are helpful while you are getting the program running. *-d* kicks in some useful things for debugging—for instance, if you specify the modem to be the current terminal (so that you can play the part of the remote system), the speed is not changed. *-v* sets a verbose mode that copies every byte received from the modem to be printed out on the terminal.

Getopt( ) is used to do command-line argument processing. If you don't already have it, see the March/April 1986 issue of *Micro/Systems Journal,* where it was printed and explained in the C Forum.

select_modem( );

Some systems have more than one modem on them, and so you have to choose one. As printed, the constants are set for three modems named /dev/tty4, /dev/tty5, and /dev/tty6. If all you have is COM1:, then change the *defines* to:

```
#define MODEM_FIRST 1
#define MODEMS 1
#define MODEM_MASK "COM%d:"
```

setup_modem( );

Once you have selected a modem, you must set the appropriate line characteristics. In particular, you want to turn off buffering, character translations, and so on. As printed, the code does this for UNIX.

For MS-DOS, the code is:

```
/* raw mode */
setmode(fd,O_BINARY);
/* 1200 baud */
sprintf(buffer,"MODE %s1200",
        modem);
system(buffer);
```

The Naval Observatory also requires even parity. Some systems can generate even parity automatically, but mine doesn't, so I have taken the hard way out. *Even( )* takes a character and returns it with even parity. *Even( )* is called from *my_write( )*, which is called upon to do all the actual output from the program.

dialup( );

Now you are ready to dial up the Naval Observatory. This requires that the system handshake with the modem in order to send the phone number to the system and make sure it connects properly. As written, the program expects to talk to a Rixon

Microsoft Avoids Challenge

We challenged Microsoft to a C compiler duel-to-the-finish, comparing compile, link and execution times, and we offered to stop advertising for two months if they won...

by Roy Sherrill, President, Datalight

Microsoft purchased our C-compiler during February 1987 and we still haven't heard from them. OK, Microsoft, we are extending our challenge deadline from April 1, 1987 to May 15, 1987. After all, the Microsoft ad claims "the fastest C you've ever seen." Your reply, Microsoft!

Walter says Optimum-C is better

Walter Bright, the developer of Optimum C, says that Optimum C would win 7 out of 10 benchmarks as compared to Microsoft C, V.4.0. Walter explained to me that Optimum C includes a unique global optimizer that helps create compact code while increasing execution speed up to 30%. By the way, Borland, Walter is still waiting for his copy of Turbo C,® V.1.0. Borland's ad claims "the fastest, most efficient and easy-to-use C compiler at any price."

After reviewing Borland's benchmarks, Walter claims that Optimum C is faster. And, as for ease of use, all Datalight C compilers have been shipped with a free Learn C program for the last six months. Also, our new EZ Interactive Editor will show you each syntax error in your source code, then compile or "make" and run your program, all from within the editor. OK, so let the Microsoft challenge begin...

We only ask the following...

The benchmark suite will consist of the set of programs that Microsoft supplied to *Computer Language* for their February 1987 C compiler review issue. Microsoft will make available the programs to Datalight at least two weeks prior to the benchmarking. The benchmarking will be between Microsoft C 4.0 and Optimum-C. It will occur at a mutually agreed upon time and place. Interested individuals will be allowed to attend. The benchmarks will be compiled and run on a standard IBM PC-AT.

There will be two separate tests for each program: compile and link speed, and execution speed. For each test, a representative from each company will set up the compiler so that it performs at its best.

The benchmarks will be adjusted so that they take sufficiently long to run, that the tolerance involved in timing them is insignificant. The winner is determined by the compiler with the faster execution times for the majority of the benchmarks. We'd like an answer from Microsoft no later than May 15, 1987.

So what's a global optimizer?

A global optimizer looks at an entire function at once, analyzing and optimizing the whole function. A technique called data flow analysis is used by Optimum-C to gather information about each function. This enables your compute-bound programs to execute as much as 30% faster after global optimization. But, there is one catch...because the global optimizer ruthlessly searches for ways to speed-up execution speed and minimize memory usage, it has relatively slow compile times. No need to worry, though, because you can merely turn the global optimizer off. In fact, you can select all, none, or part of the following optimizations: constant propagation, copy propagation, dead assignment elimination, dead variable elimination, dead code elimination, do register optimizations, global common subexpression elimination, loop invariant removal, loop induction variables, optimize for space, optimize for time, and very busy expressions.

Choose from five memory models

Speed your programs by selecting the memory model that best suits your application.

Model	Code	Data
Compact	64k total code & data	
Small	64k	64k
Program	1M	64k
Data	64k	1M
Large	1M	1M

Compiling, one step...

Now with the one step DLC program you can create .OBJ, .EXE and .COM files. Also, DLC can handle multiple files and run MASM on your assembly files.

Try Optimum-C risk free

Try Optimum-C for 30 days and if you are not 100% satisfied return it for a full refund. Also available is a C tutorial which is a combination workbook and floppy disk to help lead you through the C language with tutorials, quizzes, and program exercises.

O.K. Microsoft, it's up to you. We've put two months of advertising on the line that says you can't beat Optimum-C to a real test. Your answer, please?

PRICES

Developer's Kit w/ C Tutorial	$ 99
Optimum-C w/ C Tutorial (both with library source)	$139

Add $7 for shipping in US/$20 outside US
COD (add $2.50)

Not Copy Protected

ORDER TOLL-FREE TODAY!

1-800-221-6630

ATTENTION OEMs!

Contact us regarding arrangements.

Microsoft and MS-DOS are registered trademarks of the Microsoft Corporation. Turbo C is a registered trademark of Borland International.

Magazine Reviewers Shocked by DATALIGHT's Performance...

"Reviewing this compiler was quite a surprise for us. For such a low price, we were expecting a "lightweight" compiler. What we got was a package that is as good as or better than most of the "heavyweights." Datalight C implements a complete C language. It also compiles quickly, doesn't take up much disk space, and looks impressive in the benchmarks."

DR. DOBBS, August 1986

"This is a sharp compiler!... what is impressive is that Datalight not only stole the compile time show completely, but had the fastest Fibonacci executable time and had excellent object file sizes to boot!"

COMPUTER LANGUAGE, February 1986

Optimum-C Version 3.0

NEW!
EZ Interactive Development Environment

NEW!
Inline 8087/80287 Math Support

- Full UNIX System 5 C language plus ANSI extensions
- Fast/tight code via powerful optimizations including common sub-expression elimination
- DLC one-step compile/link program
- Multiple memory model support
- UNIX compatible library with PC functions
- Compatible with DOS linker and assembler
- Third-party library support
- Automatic generation of .COM files
- Supports DOS pathnames, wild cards, and Input/Output redirection
- Compatible with Lattice C version 3.x
- Interrupt handling in C
- Debugger support
- ROMable code support/start-up source

MS-DOS® Support Features

- Mouse support
- Sound support
- Fast screen I/O
- Interrupt handler

MAKE Maintenance Utility

- Macro definition support
- MS-DOS internal commands
- Inference rule support
- TOUCH date manager

Tools in Source Code

- cat—UNIX style "type"
- diff—Text file differences
- fgrep—fast text search
- pr—Page printer
- pwd—Print working directory
- wc—Word count

Datalight

17505-68th Avenue NE, Suite 304
Bothell, Washington 98011 USA
(206) 367-1803

R212A modem. It is trivial to convert to any other modem—for example, the first *#define* for a Hayes-compatible modem would be:

```
#define MODEM_WAKEUP  "AT\n"
```

Note that the phone number for the Naval Observatory is defined as:

```
#define NAVO_PHONE "2026531079"
```

You may need to modify this number, if you are, for example, using a WATS line or a PBX.

Dialup( ) is just one of several subroutines in this program that needs to handshake with the remote computer. This means that the program will send out a string and wait for a particular response. For example, the first thing you have to do is get an acknowledgment back from the modem that it is OK. Because this kind of "send-expect" pattern occurs many times in this program, I have written a subroutine called *waitfor( ). Waitfor( )* takes two arguments—the first is a string to expect from the modem; the second argument is the number of seconds to wait. If too much time goes by and the expected string is not seen (or too many characters go by), you return FALSE.

Waitfor( ) resembles one of the functions I covered in the September/October 1986 C Forum on reading input. I have supplied a solution that is portable to any version of UNIX. If your system does not implement timed-reads using *alarm( )*, you may have to resort to polling, as I discussed in the above column. This makes the code much simpler at the expense of using CPU time while waiting for characters to arrive.

login( );

This routine does some more handshaking, but with the Naval Observatory rather than the modem. The Naval Observatory lets anyone use its system; however, you must supply a name, organization, and where you are calling from, separated by three slashes. I use:

```
#define IDENTITY
   "DON/MSJ/GAITHERSBURG\n"
```

Please change this string (if only to keep the statistics correct).

gettime( );

Some more handshaking occurs here to

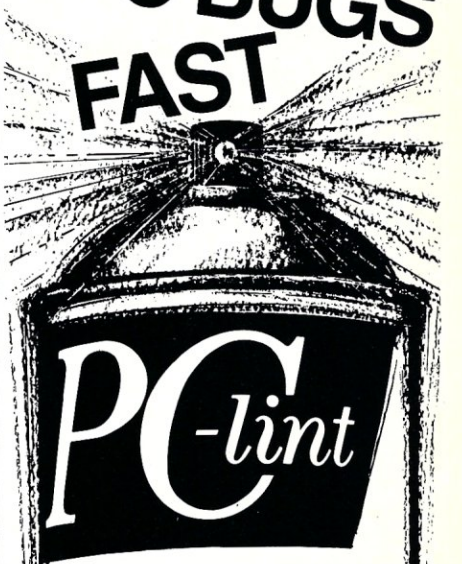
Listing 1

```
#include <stdio.h>
#include <ctype.h>
#include <getopt.h>

#define TRUE                        1
#define FALSE                       0
#define LINE_BUFFER_LENGTH 80
#define MULTILINE_BUFFER_LENGTH     (80*24)
char input[MULTILINE_BUFFER_LENGTH];

/* constants for selecting a modem */
#define MODEM_FIRST         4
#define MODEMS              3
#define MODEM_MASK          "/dev/tty%i"
/* constants for dialing a Rixon R212A modem */
#define MODEM_WAKEUP                "\r\r"
#define MODEM_PROMPT                "$ "
#define MODEM_PROMPT_TIMEOUT        20
#define MODEM_ONLINE                "7"
#define MODEM_ONLINE_TIMEOUT        45
#define MODEM_BUSY                  "5"
#define MODEM_NUMBER_PREFIX         "<k"
#define MODEM_NUMBER_SUFFIX         "\r>"

#define READWRITE 2

/* constants for talking to the Naval Observatory */
#define NAVO_PHONE                  "2026531079"
#define NAVO_LOGIN_REQUEST          "PLS IDENTIFY"
#define NAVO_LOGIN_REQUEST_TIMEOUT  60
#define NAVO_REQUEST_TIME  "@TIM\r"
#define NAVO_TIME_INDICATOR         "YYYY"  /* first string sent out after
                actual time */
#define NAVO_TIME_INDICATOR_TIMEOUT 10
#define NAVO_LOGOUT                 "@BYE\r"
#define NAVO_PROMPT                 "*\r\n"
#define NAVO_PROMPT_TIMEOUT         10
#define IDENTITY           "DON-LIBES/MSJ/GAITHERSBURG\n"

int timed_out;              /* true if the read() interrupted by alarm */
int set_time = FALSE;       /* true if we are going to try and set
               system time */
int verbose = FALSE;        /* true if conversation should be printed */
int debug = FALSE;          /* this implies verbose */

char modem[LINE_BUFFER_LENGTH] = "";         /* name of modem in /dev */
int fd = -1;                                 /* fd corresponding to modem */

/* insert main() (from article text above) here */

options(argc,argv)
int argc;
char **argv;
{
    while (EOF != getopt (argc, argv, "dvsl:")) {
        switch (optopt) {
                case 'd':
                        debug = TRUE;
                        verbose = TRUE;
                        printf ("debugging enabled\n");
                        break;
                case 'v':
                        verbose = TRUE;
                        break;
                case 'l': /* line */
                        strcpy (modem, optarg);
                        break;
                case 's': /* set system time */
                        set_time = TRUE;
                        break;
                default:
                        printf ("unknown option: %c (%x)\n", optopt,
                                optopt);
                        printf ("usage: nava [-dvls]\n");
                        exit ();
                }
        }
}

select_modem()
{
        int i;
```

```
        if (strcmp(modem,"")) {             /* user has specified modem */
                if (0 > (fd = open(modem,READWRITE))) {
                        printf("failed to open %s",modem);
                        perror("open");
                        cleanup();
                }
        } else {
                for (i=MODEM_FIRST;i<MODEM_FIRST+MODEMS;i++) {
                        sprintf(modem,MODEM_PREFIX,i);
                        if (debug) printf("trying %s\n",modem);
                        if (0 < (fd = open(modem,READWRITE))) break;
                }
                if (fd < 0) {
                        printf("no modems available\n");
                        cleanup();
                }
        }
}

/* following two functions are UNIX specific.  See article text for how */
/* to rewrite for other systems */
#include <sgtty.h>
struct sgttyb ttybuf;

setup_modem()
{
        ioctl(fd,TIOCGETP,&ttybuf);
        ttybuf.sg_flags |= RAW;                                 /* raw */
        ttybuf.sg_flags &= ~ECHO;                       /* no echo */
        /* for debugging purposes, let us use /dev/tty, but don't change */
        /* the speed! */
        if (strcmp("/dev/tty",modem)) {
                ttybuf.sg_ispeed = ttybuf.sg_ospeed = B1200;  /* 1200 baud */
        }
        ioctl(fd,TIOCSETP,&ttybuf);
}

cleanup()
{
        if (fd >= 0) {
                ttybuf.sg_flags &= ~RAW;
                ttybuf.sg_flags |= ECHO;
                if (strcmp("/dev/tty",modem)) {     /* see above */
                        ttybuf.sg_ispeed = ttybuf.sg_ospeed = B9600;
                }
                ioctl(fd,TIOCSETP,&ttybuf);
        }
        close(fd);
        exit(0);
}

dialup()
{
        my_write(fd,MODEM_WAKEUP,strlen(MODEM_WAKEUP));

        if (!waitfor(MODEM_PROMPT,MODEM_PROMPT_TIMEOUT)) {
                printf("failed to establish connection with modem\n");
                printf("make sure one is connected to %s\n",modem);
                printf("If it is, try resetting it.\n");
                cleanup();
        }

        my_write(fd,MODEM_NUMBER_PREFIX,strlen(MODEM_NUMBER_PREFIX));
        my_write(fd,NAVO_PHONE,strlen(NAVO_PHONE));
        my_write(fd,MODEM_NUMBER_SUFFIX,strlen(MODEM_NUMBER_SUFFIX));

        printf("dialing Naval Observatory...\n");

        if (!waitfor(MODEM_ONLINE,MODEM_ONLINE_TIMEOUT)) {
                printf("failed to connect to Naval Observatory\n");
                printf("last message: %s\n",input);
                printf("reason: ");
                switch (input[i] & 0x7f) {
                case MODEM_BUSY[0]:
                        prinf("busy.  Try again later.");
                        break;
                default:
                        printf("unexpected response <%c>\n",input[i]);
                        break;
                }
                cleanup();
        }
}
login()
```


#1 Lint for MS-DOS

KILLS C BUGS FAST



PC-lint

The professional diagnostic facility for C

PC-lint lets you zap swarms of C bugs and glitches at a time.

Now you can uncover the quirks, inconsistencies, and subtle errors that infest your C programs . . . waiting to bite you. PC-lint finds them all . . . or as many as you want . . . in one pass. Set PC-lint to match your own style.

Outperforms any lint at any price

- **Full K&R support and common ANSI enhancements (even MS keywords)**
- **Finds inconsistencies (especially in function calls across multiple modules!)**
- **Modifiable library descriptions for 8 popular compilers**
- **Super fast, one-pass operation**
- **Suppress any error message**
- **Zillions of options**

PRICE $139 • MC • VISA • COD

Includes USA shipping and handling. Outside USA, add $15. In PA add 6%.

ORDER TODAY, 30-day guarantee

Runs under MS-DOS 2.0 and up, and AmigaDOS. Uses all available memory.

Trademarks: PC-lint (Gimpel Software). MS, MS-DOS (Microsoft), Amiga (Commodore)

GIMPEL SOFTWARE

3207 Hogarth Lane,
Collegeville, PA 19426

(215) 584-4261

get the time. The Naval Observatory can actually provide much more information than just the time. If you are interested in the phase of the moon, times of the tides, and so on, this is the place to get it.

When the time is returned, it is printed out followed by a format description. After seeing the format description go by, you go to the previous line and pick up the date and time.

settime( );

Now you can set the time. Unfortunately, all machines set dates (and store dates for that matter) in their own ways. Rather than deal with internal system interfaces (such as DOS calls), it is easier simply to call *system( ). System( )* executes the arguments as if they had been typed in to the command processor. As printed, the program sets time for DOS systems.

DOS requires the day and month to be supplied separately, rather than as the "day of the year" given to you by the observatory. I use the function *month_day( )* from *The C Programming Language* by Kernighan and Ritchie (page 104).

The Naval Observatory returns the time relative to Greenwich, England. Most systems keep Greenwich Mean Time (GMT) internally, converting it to the local time zone upon output. If you keep local time internally, you will have to add an offset to the time from the observatory before using it. Also, be careful that you correct the day, month, and year if you and Greenwich are in different days.

logout( ); and cleanup( );

All that is left to do is to log out of the Naval Observatory's computer and hang up the phone. §

Don Libes is a computer scientist working in the Washington, D.C. area. He works on artifical intelligence in robot control systems.

C USERS' GROUP ISSUES DIRECTORY

The C Users' Group has generated a printed directory of its library of public-domain C source code. The library currently consists of approximately 200 disks available in a wide variety of disk formats. The library contains several thousand programs for CP/M, UNIX, and MS-DOS.

The printed directory is $10 (add $3 for foreign) and can be ordered from the C Users' Group, Box 97, McPherson, KS 67460; (316) 241-1065.

```
{
        if (!waitfor(NAVO_LOGIN_REQUEST,NAVO_LOGIN_REQUEST_TIMEOUT);
                printf("did not get request for identification\n");
                printf("instead, got %s\n",input);
                cleanup();
        }
        my_write(fd,IDENTITY,strlen(IDENTITY));
}

int year, day, hour, minute, second, msec;

gettime()
{
        if (!waitfor(NAVO_PROMPT,NAVO_PROMPT_TIMEOUT)) {
                printf("N.O. failed to prompt %s\n");
                printf("instead got %s\n",input);
                cleanup();
        }
        my_write(fd,NAVO_REQUEST_TIME,strlen(NAVO_REQUEST_TIME));
        if (!waitfor(NAVO_TIME_INDICATOR,NAVO_TIME_INDICATOR_TIMEOUT)) {
                printf("N.O. failed to respond with time\n");
                printf("instead got %s\n",input);
                cleanup();
        }

        /* back up to beginning of time indicator */
        for (i=strlen(input)-strlen(NAVO_TIME_INDICATOR);i>=0;i--) {
                if (0 == strcmp(NAVO_TIME_INDICATOR,input[i])) {
                        i = i - 2; /* skip crlf */
                        break;
                }
        }
        /* back up to beginning of actual date and time */
        for (;i>=0;i--) if (input[i] == '\n') break;

        if (6 != (sscanf(&input[i],"%d %d %d %d %d %d",
                        &year,&day,&hour,&minute,&second,&msec))) {
                printf("failed to find date in <%s>\n",&input[i]);
                cleanup();
        }
}

logout()
{
        my_write(fd,NAVO_LOGOUT,strlen(NAVO_LOGOUT));
        sleep(10);
}

settime()
{
        int month;

        month_day(year,day,&month,&day);
        if (set_time) {
                /* DOS specific */
                system("TIME %d:%d:%d.%d",hour,minute,second,msec/10);
                system("DATE %d-%d-%d",month,day,year);
        }
}
/* the following two functions are UNIX specific.  See the text for how */
/* to rewrite them for DOS or other systems */
#include <signal.h>

void alarm_handler()
{
        timed_out = TRUE;
}

/* waitfor() watches the input stream for occurrences of s */
/* it will timeout after "period" seconds */
int
waitfor(s,period)
char *s;
int period;
{
        int bytes_read = 0;         /* number of bytes read since entering */
        it slen = strlen(s);        /* length of string being searched for */

        signal(SIGALRM,alarm_handler);
        timed_out = FALSE;
        alarm(period);
        for (;;) {
                int cc;
                char *p, *start_searching = input;
```

10 Important Reasons C Programmers Use Our File Manager

1. It's written in C.
Clearly the growing language of choice for applications that are fast, portable and efficient. All of db_VISTA's source code is written in C.

2. It's fast – almost 3 times faster than a leading competitor.
Fast access that comes from the unique combination of the B-tree indexing method and the "network" or direct "set" relationships between records. A winning combination for fast performance.

3. It's flexible.
Because of db_VISTA's combination of access methods, you can program to your application needs with ultimate design flexibility. Use db_VISTA as an ISAM file manager or to design database applications. You decide how to optimize run-time performance. No other tool gives you this flexibility without sacrificing performance.
db_VISTA is also well behaved to work with most any other C libraries!

4. It's portable.
db_VISTA operates on most popular computers and operating systems like UNIX, MS-DOS and VMS. You can write applications for micros, minis, or even mainframes.

5. Complete Source Code available.
We make our entire C Source Code available so you can optimize performance or port to new environments yourself.

6. It uses space efficiently.
db_VISTA lets you precisely define relationships to minimize redundant data. It is non-RAM resident; only those functions necessary for operation become part of the run-time program.

7. Royalty free run-time.
Whether you're developing applications for yourself or for thousands, you pay for db_VISTA or db_QUERY only once. If you currently pay royalties to someone else for your hard work, isn't it time you switched to royalty-free db_VISTA?

db_VISTA™

Features
- **Multi-user** support allows flexibility to run on local area networks
- **File structure** is based on the B-tree indexing method
- **Transaction processing** assures multi-user consistency
- **File locking** support provides read and write locks
- **SQL-based db_QUERY** is linkable
- **File transfer** utilities included for ASCII, dBASE optional
- **Royalty-free** run-time distribution
- **Source Code** available
- **Data Definition Language** for specifying the content and organization of your files
- **Interactive database access** utility
- **Database consistency check** utility

File Management Record and File Sizes
- Maximum record length limited only by accessible RAM
- Maximum records per file is 16,777,215
- Maximum file size limited only by available disk storage
- Maximum of 256 index and data files
- Key length maximum 246 bytes
- No limit on number of key fields per record
- No limit on maximum number of fields per record

Operating System & Compiler Support
- **Operating systems:** MS-DOS, PC-DOS, UNIX, XENIX, UNOS, ULTRIX, Microport, VMS
- **C compilers:** Lattice, Microsoft, IBM, DeSmet, Aztec, Computer Innovations, Turbo C, XENIX and UNIX

8. SQL-based db_QUERY™
Add our new C-linkable, SQL-based, ad hoc query and report-writing companion product to provide a simple relational view of your db_VISTA applications. Without compromising speed.

9. Free tech support.
60 days of free technical and application development support for every Raima product. Of course, extended support and training classes are also available at your place or ours.

10. Upward database compatibility
Start out with file management in a single-user PC environment—then move up to a multi-user LAN or a VAX database application with millions of records. You'll still be using db_VISTA. That's why so many C programmers are choosing db_VISTA.

But don't just take our word for it.

"Raima's customer support and documentation are excellent. Source code availability and royalty-free run-time is a big plus."
Dave Schmitt, President
Lattice, Inc.

"db_VISTA has proved to be an all-round high performer in terms of fast execution, flexibility and portability, and has undoubtedly saved us much time and development effort."
John Adelus, Hewlett-Packard
Office Productivity Division

30-day Money Back Guarantee!
Try db_VISTA in your environment for 30 days and prove it to yourself. If not completely satisfied, return it for a full refund.

Price Schedule

	db_VISTA	db_QUERY
☐ Single user	$ 195	$ 195
☐ Single user w/Source	$ 495	$ 495
☐ Multi-user	$ 495	$ 495
☐ Multi-user w/Source	$ 990	$ 990
NEW:		
☐ VAX Multi-user	$ 990	$ 990
☐ VAX Multi-user w/Source	$1980	$1980

Order Now.
Put db_VISTA to work in your application program. Ordering is easy—simply call toll-free. We'll answer your technical questions and get you started. Call today.

Call Toll-Free Today!
1 (800) db-RAIMA
(800/327-2462) or
206/828-4636

MasterCard VISA

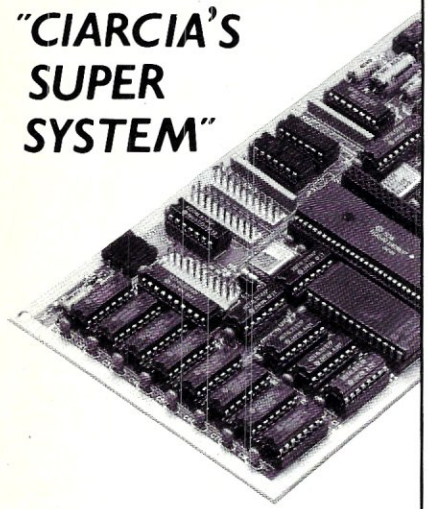
RAIMA™ CORPORATION

3055 - 112th NE, Bellevue, WA 98004 USA
(206) 828-4636 Telex: 6503018237 MCI UW

MJ57

Byte Magazine called it.

"CIARCIA'S SUPER SYSTEM"



The SB180 Single Board Computer

Featured on the cover of Byte, Sept. 1985, the SB180 lets CP/M users upgrade to a fast, 4" x 7½" single board system.

- **6MHz 64180 CPU**
  (Z80 instruction superset), 256K RAM, 8K Monitor ROM with device test, disk format, read/write.
- **Mini/Micro Floppy Controller**
  (1-4 drives, Single/Double Density, 1-2 sided, 40/77/80 track 3½", 5¼" and 8" drives).
- **Measures 4" x 7½" with mounting holes**
- **One Centronics Printer Port**
- **Two RS232C Serial Ports**
  (75-19,200 baud with console port auto-baud rate select).
- **ZCPR3 (CP/M 2.2/3 compatible)**
- **Multiple disk formats supported**
- **Menu-based system customization**

New Low Prices

SB180-1
SB180 computer board w/256K bytes RAM and ROM monitor
.....**$299.00**

SB180-1-20
same as above w/ZCPR3, ZRDOS and BIOS source **$399.00**

COMM180-S
SCSI interface **$150.00**

Now Available

TURBO MODULA-2.....$69.00
TURBO MODULA-2 with Graphix Toolbox $89.00

TO ORDER CALL TOLL FREE 1-800-635-3355

TELEX 643331

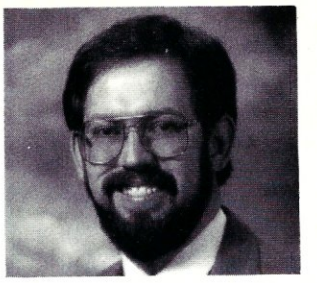
For Technical Information or in CT, call:
1-203-871-6170

Micromint, Inc.
4 Park Street
Vernon, CT 06066


```
                /* alarm can go off in code AFTER the read, in which case */
                /* we must check for it here */
                if (timed_out) return(FALSE);
                cc = read(fd,&input[bytes_read],LINE_BUFFER_LENGTH);
                /* if alarm went off, or some other signal occurred */
                /* or read failed, give up. */
                if (cc == -1) {
                        alarm(0);
                        return(FALSE);
                }

                if (verbose) write(1,&input[bytes_read],cc);
                for (p = start_searching;
                                p <= (&input[bytes_read+cc])-slen; p++) {
                        /* move over one character and try again */
                        if (debug) {
                                printf("searching for ");
                                dump(s,slen);
                                printf(" in ");
                                dump(p,slen);
                                printf("\r\n");
                        }
                        if (!strncmp(s,p,slen)) {
                                alarm(0);
                                return(TRUE);
                        }
                }
                start_searching = (&input[bytes_read+cc]) - slen;
                bytes_read += cc;
                /* if we have filled up the buffer without finding the */
                /* string, it's probably hopeless. */
                if (bytes_read + LINE_BUFFER_LENGTH >
                                MULTILINE_BUFFER_LENGTH) {
                        alarm(0);
                        return(FALSE);
                }
        }
}

my_write(fd,buf,length)
int fd;
char *buf;
int length;
{
        int i;
        int cc;

        if (verbose) write(1,buf,length);

        /* make even parity */
        for (i=0;i<length;i++) buf[i] = even(buf[i]);
        cc = write(fd,buf,length);

        if (cc != length) printf("write(,,%d) = %d?\n",length,cc);
}

dump(s,n)
char *s;
int n;
{
        int i;

        for (i=0;i<n;i++,s++) {
                *s = *s & 0x7f;
                if (isprint(*s) || *s == ' ') putchar(*s);
                else if (*s == '\r') printf("<cr>");
                else if (*s == '\n') printf("<lf>");
                else printf("<%x>",0x7f & *s);
        }
}
int
even(c)         /* return given character with even parity */
int c;
{
        int i, pbit = 0;

        for (i=0;i<7;i++) {
                if ((c>>i)&1) pbit = !pbit;
        }

        c &= 0x7f;
        if (pbit) c |= 1<<7;
        return(c);
}
```

Turbo Pascal Corner

Stephen Randy Davis



Using Turbo Pascal Chain Files

This column features tips and techniques for using Turbo Pascal productively on MS/PC-DOS and CP/M microcomputer systems. It discusses typical problems and their solutions. Reader suggestions, comments, and questions are encouraged. Address them to Turbo Pascal Corner, Route 5, Box 107K, Greenville, TX 75401 or through MCI mail, 289-6124.

In my previous column on the subject of pointers, I included some Z80 code that I had been assured would work but that I admitted having no way to test. Arthur Hendrickson, Jr., of Milwaukee wrote in with the correct *INLINE* code for CP/M-80 Turbo Pascal, Version 1.0. The entire program, including the correction, appears as Listing 1 for those who may have missed the previous issue.

I want to stress once again that using pointers to functions is a good way of avoiding *CASE* statements, which can generate a lot of code. Suppose, for example, that the user can enter a number from *1* to *5* into a program to indicate one of five different functions it is to perform. The obvious choice is to generate a *CASE* statement with five cases or four *IF* statements. The less obvious but smaller and faster choice is to create an array *RoutineAdr[1:5]* of pointers to the five different functions and *Indirect (RoutineAdr[I])* to it using the *Indirect* function introduced in the previous column. Keep this trick in mind!

I finally had the opportunity to play with Turbo Pascal for the Mac. First, Mac Turbo Pascal comes on two 3½-inch floppies: one floppy for the compiler and a second for example programs. The manual included is actually two separate manuals in one—one section describes the syntax of Turbo Pascal, much as does the "normal" manual, and the other explains how to write programs for the Macintosh interface. Both sections seem complete with plenty of examples.

This is definitely Turbo! Compilations to RAM happen virtually instantaneously; and compiling to Bernoulli disk is almost as fast. A new option called *Syntax checking* has been added, which should be even faster yet. The minimum program size appears to be roughly 10K, remarkably close to the minimum program size for CP/M and PC-DOS Turbo Pascal. Mac Turbo Pascal is syntactically identical to other Turbo Pascals as far as I can tell. Finally, Mac Turbo Pascal handles errors in the same pleasant manner as do other Turbo Pascals. I tried injecting errors into the example programs and recompiling; Mac Turbo Pascal marked the errors correctly in every case.

This is also Macintosh. One of the complaints about Borland's other products for the Mac—SideKick and Reflex—is that they do not integrate completely with the Mac interface. This does not seem to be a problem with Mac Turbo Pascal. All of the normal pull-down windows are present. Only one window gives any indication that this is a compiler at all! (Even the format window is available. Does it really make a difference whether your source program is in 12-point Helvetica with shadow or 8-point Courier?) Mac Turbo Pascal seems to have a great deal of support for the Mac interface. Several apparently built-in structures with names such as *QuickDraw* can be referenced without being explicitly defined.

If the example programs are any indication, I doubt that Mac Turbo Pascal programs will be transportable to other Turbo Pascals. The Pascal is the same, but the operating system is so different. Besides references to the built-in structures, these programs spend quite a bit of their code accessing the Macintosh hooks: defining windows, checking for mouse clicks, and so on. Mac Turbo Pascal does include the same turtle graphics routines as does PC Hopefully, Mac Turbo Pascal users will be able to adapt programs appearing in this column to their needs. I am keeping my copy of Mac Turbo Pascal around to help answer questions from Macintosh users.

TODAY'S TOPIC

Today's topic comes from Hans-Georg Joepgen of Stuttgart, West Germany. One facet of Turbo Pascal that users routinely overlook is chain files. If you look at the compile options available in Turbo Pascal, you see not only the *Compile into RAM* and *Compile to .COM file* options but also *Compile to chain file.* This is true of both PC-DOS and CP/M versions of Turbo Pascal. So what is a chain file?

A chain file is similar to a .COM file except for two things: it has the extension .CHN, and it does not include the Pascal library. You may have already noticed that if you create a program with no statements and compile it, the resulting .COM file is a whopping 10K long. This code is the Pascal library in which all of the functions built into Pascal, such as *WriteLn*, *New*, *Reset*, and so on, reside.

Unfortunately, Turbo Pascal tacks the entire Pascal library onto each and every .COM file you make, regardless of whether your program accesses any of these functions or not.

A .CHN file does not have any of this library attached. If you take a *do-nothing* program that generates 10K of .COM file and recompile it using the chain option, you find a .CHN file only 45 bytes long—quite a difference! Of course, the .CHN file cannot be executed without the Pascal library loaded into memory. Chain files are meant to be executed only from a .COM file.

Executing the .COM program loads the user code together with the Pascal library into memory. At some point the .COM file *ASSIGNs* the name of the .CHN file to a Turbo Pascal, but I doubt that any more Mac users will use these than PC users did. file variable and then invokes the built-in

Authorized Distributor Earth Computer Products

S-100 Products

Turbo Master 8	$875 $945
Turbo Slave I	$395 $410
Earth Net-100	$395 $410

PC Products

Earth Net PC	$375 $395

Prices & Shipping & Handling CIA COD

System Manager
List $375

Complete TurboDOS Network Security Package

- All Menu Driven
- Restrict Drives and User Areas
- Log ON/OFF Record Keeping
- Full Electronic Mail
- Display System Status
- Talk-Send Messages to Another Processor
- Runs on All TurboDOS Systems

We Support Large TurboDOS Systems With Multiple Masters
We Support the Following OEM's
Earth ICM ICD L/F Teletek
Dealer Discounts

Multi-User Networking Specialists

Lodden Technology Limited

2834 N. Milford Rd.
Milford, MI 48042
(313) 685-3877

TurboDOS is a registered trade mark of Software 2000, Inc.
System Manager © Lodden Technology LTD

the 64K code limit. Turbo Pascal does not support a large model as do other compilers, but it does allow programs to chain to the previous user code but leaves the Pascal library untouched. Thus, the .CHN file borrows the library of the .COM file that came before.

There are several points that should be noted, here. Pascal leaves not only the library untouched but also the global data area. (Variables declared outside any procedures are called global variables because they are accessible to all procedures.) Such variables are allocated alongside the code and not on the stack. When the .CHN file is loaded, stack variables are gone, but if the .CHN file has the *exact* same data declarations as the .COM program that came before it, these global variables will have the same values upon entering the .CHN as they did when the .COM file was exited. This allows the .COM file to pass information to the .CHN file.

.CHN files were originally intended to be Turbo Pascal's technique for breaking chain function on this variable (see the Turbo Pascal manual for details). Pascal loads the user code in the .CHN file over other code sections much as they would call a normal subroutine. The .CHN files can return to the main .COM file by using the built-in *EXECUTE* function call. Of course, loading such code sections is slow, so the number of chains should be limited as much as possible, but if the .CHN files are stored on RAM disk, the time is not too terribly significant.

Mr. Joepgen suggests you use .CHN files for something completely different. I mentioned earlier that even an empty source program generates a whopping 10K .COM file but that the .CHN file is very small. As Mr. Joepgen points out, this gives you a grand opportunity to save much of the disk space used to store your Turbo Pascal executable files. When you compile your Turbo Pascal sources, you can generate .CHN files rather than .COM files. At 10K per file, the savings add up quickly, especially if you have lots of small

Listing 1

```
Type
        address = integer;
Var
        routineadr : address;

Procedure indirect (routineadr:address; anotherarg:integer);
Begin
        INLINE (
                $FD/$E1/            {POP IY}      {get return address}
                $2A/routineadr/     {ld hl,routineadr}
                $E5/                {push hl}
                $2A/anotherarg/     {ld hl,anotherarg}
                $E5/                {push hl}
                $FD/$E5/            {push IY}     {restore return address}
                $2A/routineadr/     {ld hl,routineadr} {no go there}
                $E9                 {jp (hl)}
                )
End;

Procedure Target (dummy:address; anotherarg:integer);
Begin
        WriteLn ('We made it to the target procedure');
        WriteLn ('2nd arg = ',anotherarg:4)
End

Begin   {main}
        routineadr := addr(target);
        WriteLn ('Here we go!');
        Indirect (routineadr,2);
        WriteLn ('We made it back OK also')
End.
```

Listing 2

```
Program S;
{$p1}

{The program S ("Starter") is used to invoke CHN-files.  Before recompiling
 this program, be sure and see the Minimum Code and Minimum Data options to
 the largest possible values to allow as much room as possible for the
 chain files.

 The Definition of "p" below depends on the operating system used:

                  in CP/M-80 and MP/M II'... p: styp absolute $80;'
```

Pascal utilities.

Of course, you will need one, but only one, .COM file to execute the others. All that program does is read the next argument and the name of the .CHN file to be executed and chains to it. It is worth noting that all you are really doing here is bundling up the Turbo Pascal library in one executable file, rather than copying it individually into each little utility. Mr. Joepgen's program, which he calls S, appears as Listing 2.

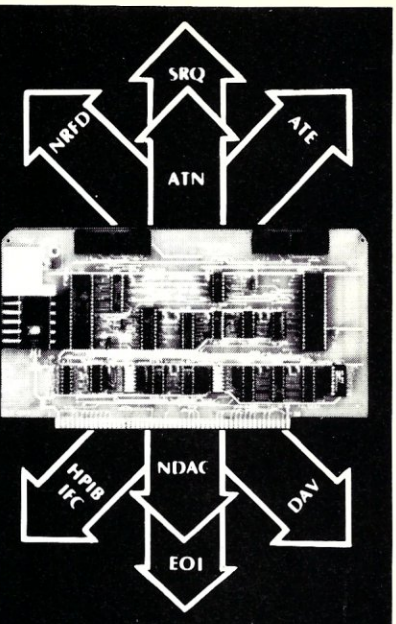
S.PAS reads the entire command line. The main program assumes that the first argument is the name of the .CHN file to execute. It copies everything after the *S* in the command line into location *CSEG:80,* where the program will expect its own arguments. It then passes the first argument to the procedure *StartVersuch* (The program was originally in German, but I have translated the rest of it myself), which provides the extension .CHN if not already present and chains to the program. Upon completion, *IORESULT* is checked to make sure all went well. If not, an error message and number are printed.

Using this technique, Turbo Pascal executable programs can be saved in substantially less disk space, which is especially helpful on floppy-based systems. Write your Turbo Pascal programs exactly as before, but just compile them using the .CHN option rather than the .COM option. Keep S.COM in your *PATH* or copy it to your CP/M system floppy so it's always at the ready. Finally, remember to tack *S* in front of your Pascal programs when executing them. Before long you will wonder how you ever got along without it.

Hans is "Georg"ing himself on those public-domain utilities, and you can too if you're the next to submit a worthy Turbo Pascal trick, suggestion, or question. §

Stephen Randy Davis is a senior systems programmer for a defense contractor in Greenville, Texas, where he programs various microprocessors. He is also working on his Masters in physics.

```
                in CP/M-86              '... p: styp absolute dseg:$80;'
                in MSDOS                '... p: styp absolute cseg:$80;' }

type styp = string[80];

var n:      integer;
    p:      styp absolute cseg:$80;
var s, ss: styp;
    i:     integer;

procedure startversuch (s: styp);     {chain to the name is 's'}
const tafel: array [0..15] of char = '0123456789ABCDEF';
var r: byte;
    f: file;
begin
  for r:= 1 to length (s) do s[r]:= upcase (s[r]); {make sure uppercase}
    if pos ('.', s) = 0 then s:= s + '.CHN';       {provide CHN extension}
  assign (f, s);
  {$i-} chain(f) {$i+};
  r:= ioresult;
    if r <> 0
    then
      begin
        writeln;
        writeln ('--- Programm "', s, '" could not be started.'^g);
        writeln ('    (Error number: $', tafel[r div 16], tafel[r mod 16], ')')
      end;
end;

begin                          {***main program***}
  n:= paramcount;
  s:= paramstr (1);
    if n > 1                   {if there is more than just the file name...}
    then begin
           ss:= '';
           for i:= 2 to n do {...concatenate all remaining args into...}
             begin             {...one string, 'p', where args are expected}
               ss:= ss + paramstr (i);
                 if i < n
                 then ss:= ss + ' ';
             end;
           p:= ss
         end
    else p[0]:= char (0);      {if no args, stuff a terminator}
  startversuch (s)             {now invoke the chain file}
end.

teLn
```

THE
488+3
IEEE 488 TO S-100
INTERFACE

S-100 ⟺ 488

- *Controls IEEE 488 (HP1B) Instruments with an S-100 computer*
- *Acts as controller or device*
- *Basic and assembly language drivers supplied*
- *Meets IEEE 696 specification*
- *Industrial quality burned in and tested up to 125K bytes/sec under software control 3 parallel ports (8255-5)*
- *$375*

D&W DIGITAL, INC.
20655 Hathaway Avenue
Hayward, California 94541
(415) 887-5711

Concurrent DOS 5.0 XM

DRI's latest version of a true multiprocessing, multiuser operating system with a high degree of MS-DOS compatibility, windowing, and lots more.

by Alex K. H. Soya

Digital Research Inc. has recently released Concurrent DOS 5.0 XM (eXpanded Memory). This latest release of Concurrent DOS introduces many new features, compared to earlier versions, and the ability to take advantage of expanded-memory cards on the IBM PC, PC/XT, and PC/AT series of computer systems. Concurrent DOS is also available for several other computer systems as implemented by their manufacturers—CompuPro, Lomas Data Products, Nixdorf, Fujitsu, Slicer, and many other companies offer versions of Concurrent DOS as the operating system for their hardware.

Concurrent DOS is a multitasking and a multiuser operating system for the Intel 8086 (8088, 80186, 80286, 80386) series processor-based computer systems. What makes Concurrent DOS different from other multitasking environments on a PC is that Concurrent DOS is its own independent operating systems, designed to be multitasking and multiuser from the ground up, rather than attempting to force the single-user PC-DOS (MS-DOS) operating system to perform multitasking operations. Concurrent DOS can thus offer features normally found only in mainframe operating systems.

Concurrent DOS allows several applications to run simultaneously on the same console or on several consoles. In a single-user environment, such as on a PC, multiple applications are run on a single console with the aid of *virtual consoles*. Virtual consoles allow a single physical console to assume different personalities. Through the use of function keys, users can switch from one virtual console to another, and windows can be set up to allow users to view several applications simultaneously as they execute concurrently.

WINDOWS AND CONSOLES

Concurrent DOS provides a window manager that allows the contents of a window, or an entire virtual console, to be written to a file. The contents of that file can later be read into other applications. It is thus possible to transfer information from one application, say a spreadsheet, to another application, perhaps a word processor, even though these applications were not designed to be compatible with each other.

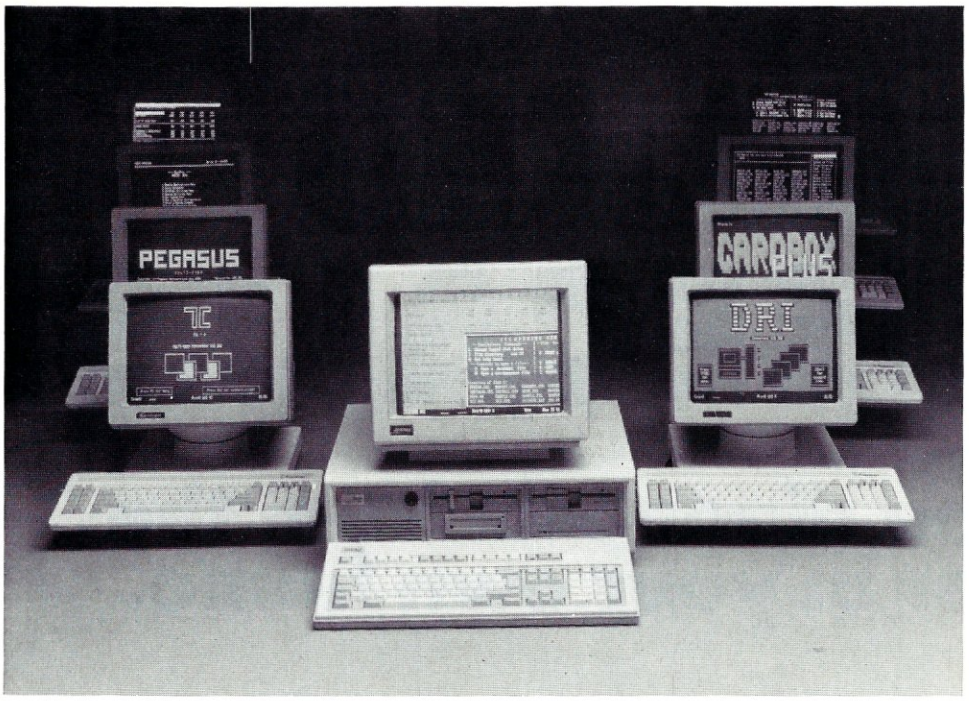
You can use the window manager to set up various window attributes, such as the window size and location, foreground and background colors, and cursor-tracking modes. These window attributes can also be set up through batch files when the system is booted or when a different window configuration is desirable for certain applications.

Unlike other windowing environments, Concurrent DOS allows applications to run simultaneously in separate windows—for example, while you are using a word processor on one virtual console, a spreadsheet can perform calculations in another window, a mail-merge program can be running in a third window, and a fourth window can be monitoring incoming calls from a modem. All the applications can run simultaneously and do not halt when you switch to different consoles.

The standard configuration of Concurrent DOS, as shipped by Digital Research Inc., provides four virtual consoles on the PC video display and allows you to connect two additional serial terminals to the PC's serial ports. Other implementations of Concurrent DOS allow as many as eight serial consoles to be added to PC or AT systems. CompuPro allows 12 or 14 physical consoles, one a PC-VIDEO card, with each physical console having several virtual consoles—up to 56 virtual consoles may be implemented on a CompuPro S-100 system, for example.

EXPANDED MEMORY

One of the most desirable features of Concurrent DOS XM is its ability to address more than 1 megabyte of memory. Through the use of EEMS (Enhanced Expanded Memory Specification) memory boards, such as the AST RAMPAGE card or the Quadram EEMS card, as much as 8 megabytes of memory are available to Concurrent DOS XM on a PC or AT.



The multitasking capability of Concurrent PC DOS Expanded Memory (XM) allows users to simultaneously run popular applications. Concurrent PC DOS XM can be customized for either single user or multiuser microcomputer systems through easy to use configuration utilities.

Other systems that use a different memory-management scheme can provide up to 16 megabytes of usable memory.

The expanded-memory feature does not require an 80286 CPU, which you might at first expect. Concurrent DOS XM can run on an 8088-based system and still access more memory than the CPU can address directly. EEMS, and other methods used by manufacturers of systems that are not PC compatible, uses a memory banking mechanism to page memory in and out of the 8086 addressing range. The EEMS method of banking memory was developed jointly by Ashton-Tate, Quadram, and AST Research. The EEMS is also often referred to as the AQA specification. An earlier method to bank memory on PC systems was developed by Lotus, Intel, and Microsoft. Also referred to as EMS (Expanded Memory Specification), this earlier specification was very limited as only up to four 16K pages (64K) could be accessed at a time. The EEMS method allows as many as 64 16K pages to be accessed simultaneously.

Several applications, such as Lotus 1-2-3 and Symphony 1.1, were specially written to take advantage of expanded memory under PC-DOS. Concurrent DOS XM uses a different approach to make use of the expanded memory. The application does not need to know about the newly available memory. Concurrent DOS XM performs its own memory management and assigns pages of memory to the application programs. At every system tick (every 1/60th of a second), Concurrent DOS XM performs a task switch. At this time the memory pages allocated to the program being switched from are disabled and the pages belonging to the program being switched to are enabled. Thus several large programs can access the entire addressable memory space simultaneously. The expanded-memory management is totally invisible to the application program.

Applications, such as Lotus 1-2-3, that were specially written to take advantage of expanded memory can still work under Concurrent DOS even though they perform their own memory management. Concurrent DOS intercepts the direct page requests from such applications and thus is aware of which pages are used by the application directly.

To gain optimum performance from Concurrent DOS XM, you should install an EEMS expanded-memory card (EEMS not EMS!), such as the AST RAMPAGE or Quadram EEMS memory card, in your system. Concurrent DOS XM can also work in 640K of conventional memory, without any expanded memory. For many users, 640K of conventional memory will be sufficient; however, those who plan to run several large applications should invest in the expanded memory.

INSTALLING CONCURRENT DOS

The installation procedure for Concurrent DOS XM varies between manufacturers. The stock DRI version for IBM PCs and compatibles installs more or less totally automatically. At installation time, you can choose from a series of easy-to-understand menus among different installation options. You can decide whether to boot Concurrent DOS from a hard or floppy disk. If you choose a hard disk, Concurrent DOS will either boot directly from the hard disk or ask if Concurrent DOS or PC-DOS should be booted after the machine is reset. You can also partition the hard disk into two different types of media—DOS media and CP/M media. On DOS media Concurrent DOS creates a special subdirectory to contain the Concurrent DOS system, including all its utilities.

Extensive setup facilities are provided to fine-tune the operating system for a particular set of hardware. The SETUP utility allows the configuration of disk-drive parameters (even items such as the step rate, head load time, and so on), memory usage (RAM drive), serial console/port configuration, clock calibration (for systems running at high CPU speeds), and some other items. Thus Concurrent DOS can be configured easily for clone systems, which

THE DOCTOR MAKES HOUSECALLS!

Get the diagnosis from the Doctor in your own home.

Subscribe to *Dr. Dobb's Journal* and enjoy the convenience of having your personal copy delivered to your home or office each month.

And you'll save over $5 off the cover price!

Every issue of *Dr. Dobb's* will bring you indispensable programming tools like algorithms, coding tips, discussions of fundamental design issues, and actual program listings. You'll find regular coverage of:

- Popular languages such as C, Assembly, Forth, Pascal, Ada, Modula-2, BASIC, FORTRAN, and Cobol.
- 68000 and 80x86 architectures
- The MS-DOS, and Unix operating systems
- Usable techniques and practical applications of AI and object-oriented programming research.
- New product reviews, and lively discussion of professional issues in software design.
- Compilers, cross assemblers and much more!

Dr. Dobb's Journal of Software Tools . . . the magazine that has lived up to its reputation as the foremost source of technical tools since 1976. One year (12 information-packed issues) is just $29.97—to subscribe simply mail in the attached card. But do it today . . . you won't want to miss *any* of the exciting issues we have planned.

may use disk drives other than those supplied with stock IBM systems. The SETUP utility is menu-driven and easy to use.

DOCUMENTATION

Concurrent DOS XM comes with a reference guide and a users' guide. The reference guide contains a detailed explanation of each available command and lists them in alphabetical order for easy indexing. The users' guide contains explanations of basic features and detailed explanations of the applications provided.

The documentation is oriented toward users rather than programmers. The explanations are usually clear and easy to understand; however, more technically minded users could have been presented with some more information, such as a more detailed description of the COMSIZE, ADDMEM, and VSET commands, all of which are used to tune the system for particular PC-DOS applications.

Although Concurrent DOS XM provides no documentation for programmers, DRI offers the Programmer's Set, which contains a programmers' manual and a system guide. Development tools are available in the form of a Developer Kit, which contains the Programmer's Set and the latest versions of the RASM86 assembler for both native-mode and PC-DOS-mode application environments.

INCLUDED APPLICATIONS

Concurrent DOS XM includes several applications. A file manager (FM) allows a completely menu-driven operation of the system. With FM, you can move around the file system, examine subdirectories, execute commands, configure the system, erase and copy files, back up the file system, and perform many other tasks.

CARDFILE is a simple electronic card storage and retrieval system that allows you to enter items by first and last name, business name, address, phone number, and a comment. You can retrieve items by selecting a search field and letting CARDFILE perform the search (such as by name) or sequentially. This application is driven by function keys, allowing fast operation.

DREDIX is the editor supplied with all DRI operating systems at the present time. This is a full-screen editor with some advanced features such as multiple-file editing and windows. An on-line help facility makes it easy to find the various commands. DREDIX is certainly a vast improvement over the days when ED was shipped with CP/M-80 and CP/M-86.

A complete menu-design and -execution system is provided by the commands EDITMENU, RUNMENU, and COPYMENU. The menus can be executed from a BATCH file or manually. The menu editor allows the use of graphic symbols, making it relatively easy to design appealing and easy-to-understand menus. With the aid of the menu manager, system integrators can set up Concurrent DOS to be completely menu-driven and remove users totally from the command line—ideal for an office environment.

A menu system allows system integrators to set up a computer system without the need for end-users to ever see the command line.

DOS MEDIA VS.CP/M MEDIA

As mentioned previously, Concurrent DOS XM can support two types of media: DOS media and CP/M media. DOS media, as the name implies, is PC-DOS (MS-DOS) compatible. As a matter of fact, the hard disk does not need any reformatting if you select DOS media. CP/M media is compatible with the earlier CP/M and CP/M-86 operating systems. The basic difference between the two types of media is in the way a logical disk drive is partitioned into separate areas. DOS media uses a tree-structured subdirectory arrangement, whereas CP/M media partitions a drive into 16 user areas. Subdirectories are accessed in a named manner—that is, each subdirectory is given a descriptive name reflecting the contents of the directory. User areas, on the other hand, are selected by a number (0-15).

I prefer DOS media as it is much easier to locate a set of files in a named subdirectory than it is to search many meaningless user-area numbers for a set of files. CP/M media does, however, have some advantages. Three types of date and time stamping can be employed on CP/M media: Access, Update, and Creation. Only two can be active at a time and thus you must chose between Create and Update stamps. Password security on files is also an option that is available only on CP/M media. Depending on the environment, the CP/M media features can outweigh the DOS media directory structure. The hard disk can also be partitioned into two partitions—one as DOS media, the other as CP/M media.

Contrary to popular belief, there is no need for PC-DOS programs to run on DOS media and CP/M applications on CP/M media. Applications can be executed and run on either media type. Many of the applications written in Concurrent DOS native mode can make use of subdirectories as well as user areas depending on what type of media they are executed on.

NATIVE MODE VS. DOS MODE

Concurrent DOS is its own independent operating system and does not reside on top of PC-DOS, as some other multitasking packages do. Concurrent DOS therefore has two modes to execute programs: native mode and DOS emulation mode. Applications written in Concurrent DOS native mode have the .CMD file extension, whereas DOS mode applications use the familiar .COM and .EXE extensions.

In Concurrent DOS native mode, the operating system provides many advanced features typically provided by large multitasking and multiuser systems. Such features include file and record locking, queue management, semaphores, process priority assignment, memory-management functions, shared-code execution, and so on. Applications that are specifically designed to perform multiuser or multitasking activities (DATAFLEX by Data Access Corp.) are written in Concurrent DOS native mode. Also, earlier CP/M-86 applications run in native mode and thus can be modified easily to take advantage of some Concurrent DOS multitasking features.

The DOS emulation mode is used to run standard PC-DOS programs in the Concurrent DOS multitasking environment. The advanced features of Concurrent DOS are not directly available to PC-DOS applications. The DOS emulation mode is exactly what it implies—an emulation of PC-DOS. Currently, PC-DOS, Version 2.1, is emulated. Concurrent DOS XM does an excellent job of bringing PC-DOS applications to the multitasking/user environment. From testing experiences I would say that about 95 percent of all PC-DOS applications run just as under PC-DOS. Applications that have trouble are mainly TSR (Terminate and Stay Resident) type programs, which try to intercept the keyboard I/O routines and perform their own keyboard handling.

As PC-DOS programs were never intended to be run in a multitasking environment, Concurrent DOS provides several commands that allow you to fine-tune the system for a particular PC-DOS application. One of the major problems with PC-DOS applications is the object file format of the .COM and .EXE files. In a multitasking environment, it is imperative to inform the operating system of how much memory is to be used by a program. Concurrent DOS native-mode object files—.CMD files—contain a header that describes the memory requirements of the program and also the type of memory model to be used (mixed code/data, separate code/data, and so on). PC-DOS .COM files, however, have no such header, and it is thus impossible for the operating system to determine the memory requirements of the program to be loaded. A similar situation occurs with PC-DOS .EXE files. Even though .EXE files do have a header, the header usually does not provide sufficient information, or the programmer addresses portions of memory that are outside the areas defined in the .EXE file header.

Concurrent DOS provides two user commands to avoid these types of problems—COMSIZE and ADDMEM. The COMSIZE command determines how much memory is allocated to a .COM file when it is loaded. The default setting during boot time is 128K, but many applications may require more than 128K of memory to run. The ADDMEM command determines how much additional memory is allocated to the amount specified in the .EXE file header when an .EXE file is loaded for execution. The default ADDMEM setting is 64K, which is usually sufficient for most .EXE applications. At times, however, programmers will address areas far beyond the memory they have reserved, in which case a larger ADDMEM setting is required.

To determine the correct COMSIZE and ADDMEM settings, you need to experiment, unless you use an application listed in the Concurrent DOS users' guide. Once you have determined the correct settings, the best way to run the application is to build a batch file containing the COMSIZE or ADDMEM commands and the name of the application. To execute the application, you would then execute the batch file that sets the correct settings.

Other problems with PC-DOS applications include programs that attempt to modify the operating system or try to communicate with hardware directly, without informing the operating system. In the latter case, you can usually run only one application that talks to the hardware directly at one time. A native-mode application would use an operating-system function to mutually exclude processes from using a limited resource, such as certain types of hardware.

One resource that several processes cannot usually share is the video-screen RAM. Unfortunately many PC-DOS applications write to the screen RAM directly rather than using operating system calls to perform this action. Thus, if more than one program paints images to the screen RAM, the results become rather confusing. For this reason Concurrent DOS provides the SUSPEND command. When the SUSPEND command is turned on (default), a PC-DOS application will be suspended as soon as the operator switches virtual consoles. Many PC-DOS applications do behave well with respect to screen I/O and can thus be run with SUSPEND turned off.

With the aid of the preceding commands, and some others, most PC-DOS applications can be made to execute well under Concurrent DOS. If an application does not work straight away, some tuning will usually help. Of course, some programs just cannot be made to run in a multitasking environment and thus cannot be used under Concurrent DOS. This situation is rare with Concurrent DOS 5.0 XM, however, and usually applies only to programs that try to modify or hook into the PC-DOS operating system.

These sorts of problems occur in any multitasking package that allows execution of PC-DOS programs—even those packages that enhance PC-DOS and thus sit on top of PC-DOS. Microsoft has already announced that its own multitasking operating system (Advanced DOS) will not be able to run most of the current PC-DOS applications. Thus, those who want to have the luxury of running a multitasking/user computer system will have to pay a small price. The native-mode features of Concurrent DOS, however, usually far outweigh the problems encountered with certain PC-DOS applications. Also, as multitasking environments become more common and not just a luxury for demanding users, most software vendors will write their software with multitasking in mind. A good example is the popular PROCOMM (a shareware application) communications package, which allows you to configure the application for

NEW LOW PRICE!
$59.95

Eco-C C Compiler

"This is the only package we reviewed that we would be willing to call a professional tool."
Computer Language, Feb., 1985

When the review mentioned above was written, the Eco-C C Compiler was priced at $250.00. Now you can have the same compiler for a mere $59.95. And that price is complete, including a library of 120 functions, all operators (except bit fields), structures, unions, long, floats, doubles, plus user's manual. We've even included a special version of the SLR Systems assembler and linker.

Benchmarks*
(Seconds)

Benchmark	Eco-C	Aztec	Q/C
Seive	29	33	40
Fib	75	125	99
Deref	19	CNC	31
Matmult	42	115	N/A

*Times courtesy of Dr. David Clark
CNC - Could Not Compile
N/A - Does not support floating point

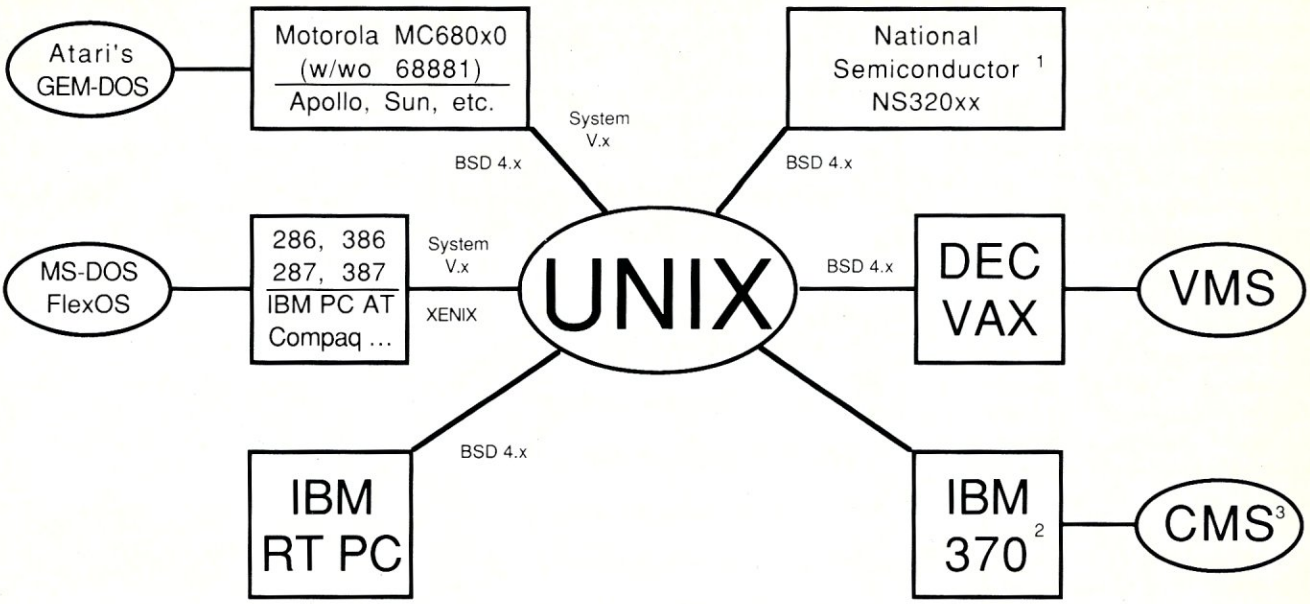
Eco-C requires 56K of free memory, 240K disk space (one or two disk drives or hard disk), Z80 CPU and CP/M 2.2 or later. We also have an MSDOS version at the same low price. Call today!

1-800-952-0472 (orders only)
1-317-255-6476 (information)

VISA MasterCard AMERICAN EXPRESS

ECOSOFT INC. (317) 255-6476
6413 N. College Ave. • Indianapolis, Indiana 46220

C and Pascal on:



We cut our teeth on UNIX, but have become famous on MS-DOS, where we out-did UNIX with our UNIX-like DOS Helper ™ utilities (find, fgrep, cat, ls, mv, tail, uniq, and wc) and our superior optimizing compilers.

Professional Pascal ™ and High C ™ are now well-respected on the PC by organizations such as Ansa, Ashton-Tate, AutoDesk, Sky, NYU, Multimate, Lifetree, Daisy, ...and Computer Language Magazine, Dr. Dobbs' Journal, PC Tech Journal, PC Magazine, and the Journal of Pascal, Ada, and Modula-2.

Our C Validation Suite will blow your C compiler out of the sea, while our C compiler tracks the emerging ANSI Standard and generates tighter code with far better lint-like feedback help.

And you'll love Professional Pascal's Ada-like packages, true data abstraction, C-like bit manipulation, and much more, along with the tight code that is linkable with High C, or other C, object modules (and vice versa).

Our Translator Writing System (TWS) picks up where YACC and LEX left off.

All uniformly implemented on UNIX, VMS, CMS, MS-DOS, FlexOS,... .

Professional software developers in need of industrial-strength tools should contact:

MetaWare™ INCORPORATED

903 Pacific Avenue, Suite 201 • Santa Cruz, CA 95060-4429
(408) 429-META (6382) • Telex: 493-0879 META UI

The Clear Choice for Large Programming Projects.

• MetaWare, High C, Professional Pascal, and DOS Helper are trademarks of MetaWare Incorporated • Others — their owners are: Ada — DoD; Apollo — Apollo Computer; Atari — Atari Corporation; dBase III — Ashton-Tate; DEC, VAX, VMS — Digital Equipment Corporation; FlexOS, GEM-DOS — Digital Research Inc.; IBM, RT PC — IBM Corporation; MS-DOS — Microsoft Corporation; Sun — Sun Microsystems; UNIX — AT&T; Volkswriter — Lifetree Software •
© 1987 MetaWare Incorporated.

Footnotes: 1. NS320xx version by special order. 2. 370/UNIX not yet available. 3. CMS version available by 7/87.

execution in a multitasking environment, in which case PROCOMM suddenly turns itself into a "well-behaved" program.

SOME BELLS AND WHISTLES

Concurrent DOS XM also introduces some new features that were not available in previous versions of Concurrent DOS. One of the most useful new features is command-line editing. The Concurrent DOS command-line syntax is similar to the PC-DOS command line. When users misspell a file name or a command, the usual action is to retype the misspelled command line, but under Concurrent DOS XM, this is no longer necessary. A command-line editor is provided that allows you to recall, edit, or reissue previously issued commands. The editor uses WordStar-like commands or the arrow keys on the numeric keypad to move around the list of previous commands and to delete, insert, or overwrite characters.

Also available is an automatic command-recall feature, which you can toggle on or off. When the feature is enabled, the operating system searches through the list of previous commands to locate a match for the command currently being typed. Thus, to reissue the same command, in most cases you need only type the first one or two letters of the command. The command line is automatically filled in with the currently recognized command, and as more letters are typed, a different command may be matched or a new command recognized.

The command-line editor has certainly sped up my moving around Concurrent DOS. Also, I no longer find myself writing batch files to perform a sequence of commands I need to reissue several times—I just use the command-line editor to recall the previous sequence of commands. This feature is most useful to me when compiling, linking, and running a new program several times while debugging.

Output and input redirection is another added feature. The syntax is identical to that of PC-DOS. Output redirection can be targeted at more than one device at a time, so the output of an application can be redirected not only to a single file but also to the console or the printer simultaneously.

A PROMPT command has finally been added to allow you to specify your favorite prompt string. The prompt may include the date, time, directory path, user number, and even special escape sequences to perform screen formatting.

One item that has needed improvement ever since the days of CP/M is the batch language processor. This has finally been accomplished with Concurrent DOS XM. The BATCH processor now supports conditionals and execution error-level support. Supported BATCH statements include ECHO, FOR, GOTO, IF, PAUSE, REM, and SHIFT. Through the use of variables and conditionals, you can create flexible batch files that can take care of almost any unforeseen situation.

AVAILABILITY

Concurrent DOS is available from several suppliers. For the IBM PC, PC/XT, and PC/AT and its large variety of 100 percent compatibles, it is available directly from DRI (call DRI telemarketing at (800) 443-4200). The current list price is $395. Other sources are available, depending on how many extra features and utilities you require.

Through the use of expanded-memory techniques, Concurrent DOS XM overcomes the limited address space of 8088/6 CPUs and the 640K barrier of PC systems.

Haar Industries offers an enhanced version of Concurrent DOS XM called MUCDOS. The MUCDOS version of Concurrent DOS comes as a three-, five-, seven-, or nine-user system in a price range from $495 for the three-user system to $1,795 for the nine-user version. MUCDOS is targeted primarily at the ITT XTRA/286 and ITT XTRA XL AT clones but is also available for other AT clones. Haar Industries is based in Washington, D.C., and can be reached at (202) 338-8550.

Goodall Computer Systems is another source for an enhanced version of Concurrent DOS XM. Goodall also provides two, four, or eight additional serial ports and is targeted at the IBM PC, PC/XT, and PC/AT. Marketed as GCS Concurrent DOS, this package includes many system utilities and additional features. A networked version, based on DRNET, is also available. Prices range from $349 for a three-user system, $395 for the system with additional serial ports, to $1,895 for a two-node network system (each additional node is $595). Goodall Computer Systems can be reached at (415) 648-2174.

A single-user version of Concurrent DOS XM is shipped with the Quadram EEMS memory card. This version is a subset of the DRI release of Concurrent DOS XM. A certificate is included with the Quadram version to upgrade to the full version of Concurrent DOS XM.

Concurrent DOS is by no means restricted to PC-type systems. As already mentioned, many of the major microcomputer manufacturers offer versions of Concurrent DOS for their systems. Among others, Concurrent DOS is available for CompuPro, Lomas Data Products, Ampro, Fujitsu, Slicer, Sword, Nixdorf, and Apricot computer systems. Contact the manufacturer of your system for availability and pricing.

IN SUMMARY

Concurrent DOS XM is a full-featured multitasking and multiuser operating system. Through the use of expanded-memory techniques, Concurrent DOS XM overcomes the limited address space of 8088/6 CPUs and the 640K barrier of PC systems. Concurrent DOS runs most existing PC-DOS applications and provides sophisticated features to applications written in native mode. To use the multiuser features of Concurrent DOS effectively, you should opt for applications written in native mode when purchasing new programs, although even the use of existing PC-DOS applications makes the multitasking feature of Concurrent DOS highly desirable.

Concurrent DOS provides a familiar user environment, as commands are similar or identical to those of PC-DOS. A menu system allows system integrators to set up a computer system without the need for end-users to ever see the command line.

Compared to previous version of Concurrent DOS, the XM version has been greatly enhanced and provides many desirable features. Owners of earlier versions should definitely consider updating to this latest release. §

Alex Soya has been involved with CP/M for more than nine years. He is currently a senior in computer science at Florida Institute of Technology. His interests include operating systems, data communications, system utilities, and real-time and scientific programming. Alex is a member of the Association for Computing Machinery (ACM).

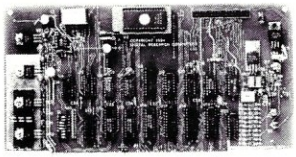
ALL SALES ARE MADE SUBJECT TO THE TERMS OF OUR 90 DAY LIMITED WARRANTY. A COPY OF THIS WARRANTY IS AVAILABLE FREE, ON REQUEST.

DIGITAL RESEARCH COMPUTERS

(214) 225-2309

S100 EPROM PROGRAMMER

OUR NEWEST DESIGN, FOR FAST EFFICIENT PROGRAMMING OF THE MOST POPULAR EPROM'S ON YOUR S100 MACHINE. COMES WITH MENU DRIVEN SOFTWARE THAT RUNS UNDER CP/M 2.2 (8 INCH). PC BOARD SET CONSISTS OF (S100) MAIN LOGIC BOARD REMOTE PROGRAMMING CARD AND SIX PERSONALITY MINI BOARDS FOR 2716, 2532, 2732, 2732A, 2764, AND 27128. SOLD AS BARE PC BOARD SET ONLY WITH FULL DOC. SOFTWARE FEATURES "FAST" PROGRAMMING ALGORITHM. FOR Z80 BASED SYSTEMS.

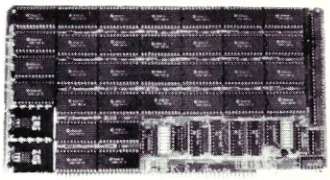


PC BOARD SET, FULL DOCUMENTATION, 8 IN. DISKETTE WITH SOFTWARE.

NEW! $69.95

64K S100 STATIC RAM

$99.00 KIT



LOW POWER!

150 NS ADD $10

BLANK PC BOARD WITH DOCUMENTATION $49.95

SUPPORT ICs + CAPS $17.50

FULL SOCKET SET $14.50

FULLY SUPPORTS THE NEW IEEE 696 S100 STANDARD (AS PROPOSED)

ASSEMBLED AND TESTED ADD $50

FEATURES: *PRICE CUT!*

- Uses new 2K x 8 (TMM 2016 or HM 6116) RAMs.
- Fully supports IEEE 696 24 BIT Extended Addressing.
- 64K draws only approximately 500 MA.
- 200 NS RAMs are standard. (TOSHIBA makes TMM 2016s as fast as 100 NS. FOR YOUR HIGH SPEED APPLICATIONS.)
- SUPPORTS PHANTOM (BOTH LOWER 32K AND ENTIRE BOARD).
- 2716 EPROMs may be installed in any of top 48K.
- Any of the top 8K (E000 H AND ABOVE) may be disabled to provide windows to eliminate any possible conflicts with your system monitor, disk controller, etc.
- Perfect for small systems since BOTH RAM and EPROM may co-exist on the same board.
- BOARD may be partially populated as 56K.

128K S100 STATIC RAM/EPROM BOARD

JUST OUT! USES POPULAR 8K X 8 STATIC RAMS (6264) OR 2764 EPROMS. FOR 8 OR 16 BIT DATA TRANSFERS! IEEE 696 STANDARD. LOW POWER. KITS ARE FULLY SOCKETED. FULL DOC AND SCHEMATICS INCLUDED. 24 BIT ADDRESSING.

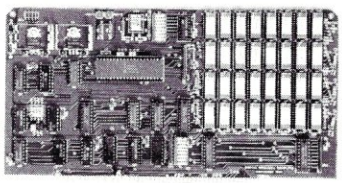
NEW!

$59.95	$219.00	$139.00
BARE PC BOARD	128K RAM KIT	128 EPROM KIT

256K S-100 SOLID STATE DISK SIMULATOR!

WE CALL THIS BOARD THE "LIGHT-SPEED-100" BECAUSE IT OFFERS AN ASTOUNDING INCREASE IN YOUR COMPUTER'S PERFORMANCE WHEN COMPARED TO A MECHANICAL FLOPPY DISK DRIVE.

PRICE CUT!



FEATURES:

- 256K on board, using +5V 64K DRAMS.
- Uses new Intel 8203-1 LSI Memory Controller.
- Requires only 4 Dip Switch Selectable I/O Ports.
- Runs on 8080 or Z80 S100 machines.
- Up to 8 LS-100 boards can be run together for 2 Meg. of On Line Solid State Disk Storage.
- Provisions for Battery back-up.
- Software to mate the LS-100 to your CP/M* 2.2 DOS is supplied.
- The LS-100 provides an increase in speed of up to 7 to 10 times on Disk Intensive Software.
- Compare our price! You could pay up to 3 times as much for similar boards.

BLANK PCB (WITH CP/M* 2.2 PATCHES AND INSTALL PROGRAM ON DISKETTE) $24.95 (8203-1 INTEL $29.95)

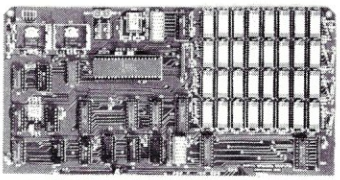
CLOSE OUT! BLANK PCB ONLY: $24.95

#LS-100

1 MEG. S-100 SOLID STATE DISK SIMULATOR!

WE CALL THIS BOARD THE "LIGHT-SPEED-100" BECAUSE IT OFFERS AN ASTOUNDING INCREASE IN YOUR COMPUTER'S PERFORMANCE WHEN COMPARED TO A MECHANICAL FLOPPY DISK DRIVE.

LS 100 II ***NEW!***



FEATURES:

- 1 Meg. on board, using +5V 256K DRAMS.
- Uses new Intel 8203-1 LSI Memory Controller.
- Requires only 4 Dip Switch Selectable I/O Ports.
- Runs on 8080 or Z80 S100 machines.
- Up to 4 LS-100 boards can be run together for 4 Megs. of On Line Solid State Disk Storage.
- Provisions for Battery back-up.
- Software to mate the LS-100 to your CP/M* 2.2 DOS is supplied.
- The LS-100 provides an increase in speed of up to 7 to 10 times on Disk Intensive Software.
- Compare our price! You could pay up to 3 times as much for similar boards.

BLANK PCB (WITH CP/M* 2.2 PATCHES AND INSTALL PROGRAM ON DISKETTE) $59.95 (8203 1 INTEL $29.95)

(ADD $50 FOR A&T) $259.00

#LS-100 II (FULL 1 M.B. KIT)

1 MEGA BYTE!

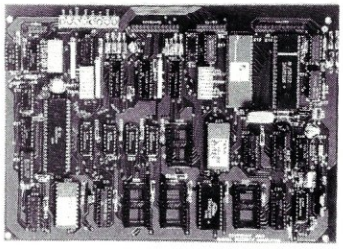
ZRT-80 CRT TERMINAL BOARD!

A LOW COST Z-80 BASED SINGLE BOARD THAT ONLY NEEDS AN ASCII KEYBOARD, POWER SUPPLY, AND VIDEO MONITOR TO MAKE A COMPLETE CRT TERMINAL. USE AS A COMPUTER CONSOLE, OR WITH A MODEM FOR USE WITH ANY OF THE PHONE-LINE COMPUTER SERVICES.

FEATURES:

- Uses a Z80A and 6845 CRT Controller for powerful video capabilities.
- RS232 at 16 BAUD Rates from 75 to 19,200.
- 24 x 80 standard format (60 Hz).
- Optional formats from 24 x 80 (50 Hz) to 64 lines x 96 characters (60 Hz).
- Higher density formats require up to 3 additional 2K x 8 6116 RAMS.
- Uses N.S. INS 8250 BAUD Rate Gen. and USART combo IC.
- 3 Terminal Emulation Modes which are Dip Switch selectable. These include the LSI-ADM3A, the Heath H-19, and the Beehive.
- Composite or Split Video.
- Any polarity of video or sync
- Inverse Video Capability.
- Small Size: 6.5 x 9 inches.
- Upper & lower case with descenders.
- 7 x 9 Character Matrix.
- Requires Par. ASCII keyboard.

FOR 8 IN. SOURCE DISK (CP/M COMPATIBLE) ADD $10



$89.95 #ZRT-80 (COMPLETE KIT, 2K VIDEO RAM)

A&T ADD $50

BLANK PCB WITH 2716 CHAR. ROM. 2732 MON. ROM $49.95

SOURCE DISKETTE - ADD $10

SET OF 2 CRYSTALS - ADD $7.50

THE NEW 65/9028 VT ANSI VIDEO TERMINAL BOARD!

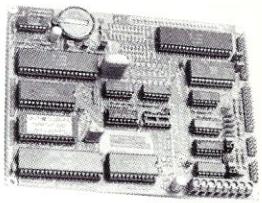
★ FROM LINGER ENTERPRISES ★

A second generation, low cost, high performance, mini sized, single board for making your own RS232 Video Terminal. This highly versatile board can be used as a stand alone video terminal, or without a keyboard, as a video console. VT100, VT52 Compatible.

FEATURES:

- Uses the new CRT9128 Video Controller driven by a 6502A CPU
- On-Screen Non-Volatile Configuration
- 10 Terminal Modes: ANSI, H19, ADM-5, WYSE 50, TVI-920, KT-7, HAZ-1500, ADDS 60, QUME-101, and Datapoint 8200
- Supports IBM PC/XT, and Parallel ASCII Keyboards
- Supports standard 15.75 kHz (Horiz.)
- Composite or Split Video (50/60 Hz)
- 25 X 80 Format with Non-Scrolling User Row
- Jump or Smooth Scroll
- RS-232 at 16 Baud Rates from 50 to 19,200
- On Board Printer Port
- Wide and Thin Line Graphics
- Normal and Reverse Screen Attributes
- Cumulative Character Attributes: De-Inten, Reverse, Underline and Blank
- 10 Programmable Function Keys and Answerback message
- 5 X 8 Character Matrix or 7 X 9 for IBM Monitors
- Mini Size: 6.5 X 5 inches
- Low Power: 5VDC @ .7A, ± 12VDC @ 20mA.

MICRO SIZE!



$99.95 FULL KIT

w/100 Page Manual

ADD $40 FOR A&T

OPTIONAL EPROM FOR PC/XT STYLE SERIAL KEYBOARD: $15

SOURCE DISKETTE: PC/XT FORMAT 5¼ IN. $15

Digital Research Computers

P.O. BOX 381450 • DUNCANVILLE, TX 75138 • (214) 225-2309

TERMS: Add $3.00 postage. Orders under $15 add 75¢ handling. No C.O.D. We accept Visa and MasterCard. Tex. Res. add 6-1/4% Tax. Foreign orders (except Canada) add 20% P & H. Orders over $50 add 85¢ for insurance.

*TM OF DIGITAL RESEARCH INC. (CALIF.)

WE ARE NOT ASSOCIATED WITH DIGITAL RESEARCH INC. (CALIF.) THE SUPPLIERS OF CPM SOFTWARE

Santa Cruz XENIX V

by Wynford Flan

True Multiuser Multitasking UNIX for the PC

UNIX ON A PC?

When the IBM PC first came out, people scoffed at the idea of running UNIX on it. No memory protection! A toy CPU! Limited memory! All these criticisms were as valid then as now, but Microsoft was in the business of porting UNIX to small machines and it was undaunted. Eventually it emerged with a version of its XENIX that ran on the 8088/8086 and hence on the IBM PC/XT. Since then, several UNIXs have become available for the PC. This article looks at XENIX as adapted by The Santa Cruz Operation (SCO).

I ran SCO XENIX System V, Release 2, on an Olivetti M24, an 8086-based PC/XT clone that is almost identical to the AT&T 6300 (not 6300 plus), with 640K of memory and one hard disk. Note that SCO XENIX System V, Release 2, is the second release of SCO XENIX System V. It is not the same as AT&T System V, Release 2—it is a merger of parts of System V with parts of Berkeley UNIX and Seventh Edition (V7) UNIX. It's common for small-system UNIX ports to have this mixed ancestry; I'll discuss a few of the significant differences between System V and SCO XENIX as I go.

DOCUMENTATION

The documentation for SCO XENIX V is a series of PC-style slipcase manuals in loose-leaf binders. The binders are slick, and the contents are printed adequately, if plainly.

The *Programmer's Reference* has the conventional manual sections but with a nonstandard nomenclature. The letter *C* is used for commands, so you see entries such as CC(C) and DOSLD(C) for commands. System services is labeled *S* and contains the material that was in Section 2 and Section 3 merged together. Other sections are similarly named with letters. Well, it's different—different from V7 and from Berkeley, which still adheres to the original numbering; different from System V, which uses a revised numbering; and different from the *System V Interface Definition*, which uses condensed names such as BU_CMD for Basic Utilities_Commands. Amusingly, the permuted index is at the back. Most books have their alphabetical index at the back, but the UNIX manuals always have the permuted index (keyword index) at the front. Because the manuals are loose-leaf, you can have it whichever way you want.

The *Programmer's Guide* contains the additional material not covered in the main pages. It has the information you need on compiler models for the 8088's baroque segmented architecture. Because the 8-bit 8088 and the 16-bit 8086 do not have memory protection, you can't dynamically expand the stack at run time, so there's a command to manipulate the stack size. All this, as well as some information on C language portability across UNIX variants, is contained in the programmer's documentation.

Other manuals contain information on using the system. They seem to be about average; I didn't use them much. The documentation is conventional UNIX quality; if you're used to any standard UNIX system, the SCO documentation won't get in the way.

THE VISUAL SHELL

For those who don't know how to use the UNIX commands yet, and for those who don't want to learn them, SCO XENIX V has a facility called the visual shell. It presents two windows—a list of files in the current directory and a list of commands. You move about with control keys and execute commands with other keys. In the Files window, for example, touching a file name with = prints the file on your terminal if the file is printable or runs it if it is runnable. The visual shell seems well enough thought out.

COMMANDS

Most UNIX commands work in the same way on all UNIX systems: V7, Berkeley, or System V. And SCO XENIX V is an amalgam of pieces from all three types of UNIX stock. Where there are differences, SCO XENIX tends to conform to the System V model. But experienced users of System V will find some surprises in the system and software tool programs.

The *file* command tries to figure out what the contents of a file are. For example:

```
file /bin/cat
```

should say something about executable, and:

```
file /
```

should tell you that / is a directory. The SCO XENIX version of this program derives from V7 rather than from System V. These two versions differ in how they learn about file types. In most traditional UNIXs, for example, a file with the number *407* octal stored in the first two (or four) bytes is recognized as an ordinary compiled program in a form ready for execution. The original *file* command had all its knowledge hard-coded in the C language source for the program; for example, it had the number *0407* stored in an array or a switch statement. The System V file command is better; the magic numbers for a range of file types are stored in a human-readable file on disk, and you can edit this magic numbers file or even provide an alternate file of magic numbers. (Knowledge about directories, special

files, and files that can be classified only by reading through the first few lines of the file is always hard-coded in the source for *file*). SCO XENIX V has the less-well-developed version of the program. If you're doing software development for a range of computers, for example, you need to be able to insert your own magic numbers into the table. Without XENIX source, you can't.[1]

The *file* command provided is adequate for most users, though. Some of the functionality is broken out into another program called dtype to determine just the contents of a floppy. Perhaps there is a good reason for this separation, related to reading the contents of a floppy versus reading about the special file (device interface) it is mounted on. Most other UNIXs keep this functionality in the *file* command.

Version 7 included numerous useful functions not found in System V. One such item is the dbm(3) library, a series of routines that provides keyed storage and retrieval. DBM retrieval is almost always faster than ISAM. SCO XENIX includes the dbm library, compiled for all three models. I didn't have a chance to try using the routines, but they're there in their rightful place.

Another way in which V7 excels over System V is with its *mv* command. There is no reason why you should not be able to *mv* a directory from one place to another under your home directory. But on System V you need to use a special kludge called *mvdir* to move directories around within a file system, and only the superuser is allowed to use *mvdir*. Sadly, the SCO XENIX *mv* conforms to System V rather than to V7 or BSD.

The standard shells—sh and csh—are both included. Csh does not have the job control facilities that it has under Berkeley systems, but neither do most System Vs. Presumably a future release will include shell layers, the System V method of task multiplexing for dumb terminals.

The standard editors ed and ex/vi are both included. Ed is the original editor that was written in the very first UNIX; it is line-oriented, and simple to learn but provides little feedback. Vi is Berkeley's screen editor; widely known and used, it is the de facto standard UNIX screen editor. Both editors behave as they would on a larger UNIX time-sharing system.

There are several line-printer spooler systems in circulation. The most important[2] are the V7 *lpr*, the Berkeley *lpr*, and the System V *lp* systems. SCO could have included any of these systems; it chose to go with the System V *lp* command. Given that decision, it took the obvious sensible step of making *lpr* an alias for *lp*. Thus programs (from a V7 or 4BSD background) that assume there is an *lpr* command will still work. But the administration commands are those of System V, so any experience you have with those commands can be applied here. There is no Berkeley-style /etc/printcap file; instead, you write a shell script to interface to any odd or new devices. To my mind the System V version is superior here, and SCO XENIX V has it.

Every system needs either a paginator program or pagination in the terminal driver. I've used both, and there is no question in my mind that the latter is superior if done well. Despite my opinions, paginators are trendy this decade, so people demand them and UNIX vendors include them. SCO XENIX V includes both Berkeley's *more* and System V's *pg*, so if you've used either of these you'll feel at home.

One problem that everybody must face in this day of binary-only systems is that of adjusting the defaults on a program you

PC-SPRINT

"PC-Sprint is the most cost effective PC Speedup product on the market."
— Computer Shopper Magazine

- Run your PC, XT or clone at 7.38 mhz.
- 280% Speedup (Norton SI rating)
- Speeds up all software — you can see the difference
- External speed switch
- External reset button
- Change speed "on the fly"
- Compatible with 8087
- Works with all color or mono displays
- "Slotless" plug-in on most PCs
- Includes: Selectable top speed, instructions, warranty, tool, remote mount switch, free BBS subscription

$89.95 V20 add $10. Call for information on other products

Exec-PC, Inc.
P.O. Box 11268 Shorewood, WI 53211
(414) 242-2173

don't get the source for. SCO XENIX V takes the sensible approach of providing an external, human-readable text file containing the default settings for certain key programs. The file /etc/default/tar, for example, contains the default settings for the tar program—not as much fun as having system source, of course, but a step toward flexibility in the world of binary-only UNIX.

To sum up, this system isn't fully System V. It has some programs taken from V7 or BSD UNIXs instead of the corresponding System V functions, but it also has functionality that is in V7 and BSD and not in System V. It does have all the important pieces—all in all, it's not a bad mix.

MULTIUSER OPERATION

All UNIXs and true derivative UNIX-alikes are inherently multiuser and multitasking. You can always multitask by putting jobs in the background—with the standard shells you just put an ampersand (&) at the end of the command. To make UNIX multiuser, you just tell init how many users you want and where you want them.

Init is the program that starts all the other tasks running when you boot the system and enables a login when a user logs off a terminal. There are two general versions of init in the world—one from V7 and one from System V. The SCO XENIX init is like that of V7 and Berkeley UNIXs—that is, it reads its list of terminal tasks to be started from the file /etc/ttys, instead of from /etc/inittab as do standard System V versions. Clearly Microsoft or Santa Cruz thought the V7-style init superior. It is perhaps better for smaller systems, but it's less general. The System V version of init offers a few run states, each one a combination of terminals enabled and disabled. It will also let you run almost any arbitrary program (such as a spooling system) that will be shut down automatically when you enter a run state that it's not defined in. The older init, used in SCO XENIX, lets you enable or disable terminals only on a per-line basis.

The format of the /etc/ttys file is the same as on V7 and Berkeley UNIXs—that is, it looks like this:

```
1mconsole
0mtty02
16tty03
```

The first column is either 1 to enable logins on that port or 0 to disable logins. The next letter (*m* or *6* in these examples; see the manual for a complete list) is an (admittedly cryptic) code for the baud rate that is to be used if login is enabled. If login is disabled, any other program using the port, such as uucp or a spooling system, would control the baud rate, and this key letter would be ignored. The third field is the port name; *console* refers to the terminal named /dev/console and *tty02* refers to /dev/tty02, and so on.

An amusing aside is that SCO XENIX V manuals describe the System V version of the routines for handling utmp and wtmp files, including information on tracing the run-state changes. But the mechanism for changing run states isn't there yet. SCO XENIX has only single-user (maintenance) and multiuser modes. A good feature is that it demands the root password before entering maintenance mode; older UNIXs (and even some newer ones, such as Sun UNIX 3.0) do not have this security arrangement.

C PROGRAMMING

The C compiler works as you'd expect for simple programs. Here is an example of compiling the world's most-used C test program and a listing to show how long it took and how big the result was:

```
$ cat t.c
main( )
{
        printf("hello world\n");
        exit(0);
}
$ time cc t.c
real     29.2
user      7.4
sys       6.6
$ a.out
hello world
$ ls -l
total 20
-rwxr-xr-x 1 Wyn  7658 Oct 5 a.out
-rw-r--r-- 1 Wyn    76 Oct 5 log
-rw-r--r-- 1 Wyn    47 Oct 5 t.c
```

The real time is what you would see with a stopwatch—about 30 seconds. The user is how much CPU was used by the user program—the compiler and loader in this case. The system time is the CPU time taken by the operating system servicing system call requests for the user. Ideally, on an unloaded system with fast disk, these times would be almost equal. With a slow hard disk, such as that found on most inexpensive PC-class machines, you would expect the elapsed time to exceed the CPU time even for a single task, as it does here. The compile time is not stunning. It is comparable to some of the oldest 68000 UNIXs (an old 68000 system, underconfigured with only 750K and a slow 45-msec disk for this test, took just over 30 seconds), though the latest 68020 boxes leave it in the dust (a Sun-3/180 with one Fujitsu Eagle disk and several users compiles the program in 3 seconds flat, with 0.6 seconds user time and 0.9 seconds system time).

Execution times for this program are not measurable. To do benchmarking is to lead you to expect that your performance will be similar. Because everybody's work load is different from everybody else's, I've steered clear of benchmarking the run times of particular programs.

But it's certainly fair to run several copies of the same program. Bear in mind that this is a worst-case test; several programmers using a system would almost never start compiles at exactly the same time. The commands:

```
$ cp t.c u.c
$ time cc -o t t.c &
$ time cc -o u u.c
```

produced the reports:

```
t.c:
real     44.5
user      7.2
system    5.5
u.c:
real     45.0
user      7.4
system    6.5
```

The user and system times don't vary much; the clock time goes up substantially but is far less than double. Running three compiles simultaneously seems to go past the limit of reasonable; the real or clock times for three compiles together were 1:49, 1:54 and 2:05. Here the elapsed time has quadrupled (from 30 to 120 seconds) when the number of tasks has only tripled. Because the system has only 640K of real memory, it has resorted to swapping the different C compilers in and out; this slows the entire system down.

MEMORY MODELS

Segmented memories are atavisms that should not be propagated; if the IBM PC didn't have its 8-bit 8088 CPU inside, we would not even hear much about 8088/8086-based systems. But it does, and these peculiar architectures are with us to stay, alas. The SCO XENIX compiler can gener-

Taming MS-DOS

by Thom Hogan

This book is an advanced user's guide to enhancing the MS-DOS environment. Picking up where you DOS manual leaves off, **Taming MS-DOS** takes you beyond the basics. Every chapter is full of tips and techniques that extend the power of DOS so you can work more efficiently.

- You'll learn to customize CONFIG.SYS to maximize the performance of your system and how to use ANSI.SYS to tailor your system prompt and monitor attributes to fit your needs.
- Learn to maximize your batch files with routines using redirection, filters and pipes. You'll find routines that prevent accidental reformatting of your hard disk, redefine function keys and locate files within subdirectories.
- You'll also find batch files that implement a DOS help system, including help text files, a menu system that interprets keyboard input, and a routine for quick redefinition of function keys.
- Taming MS-DOS includes nearly 50 ready-to-use programs that increase DOS's functionality. Now you can easily rename directories and disk volumes, change file attributes, check available RAM and disk memory, display a memory-resident clock, and assign DOS commands to ALT keys.
- Quick reference charts provide easy access to batch command syntax, CONFIG.SYS syntax and ANSI.SYS strings.

The programs, including batch files and DOS enhancements, are available on disk along with full source code.

Taming DOS **Item #060** **$19.95**
Taming MS-DOS with disk **Item #061** **$34.95**

FULL SOURCE CODE INCLUDED!

Nr: A Nroff-like Text Processor for MS-DOS

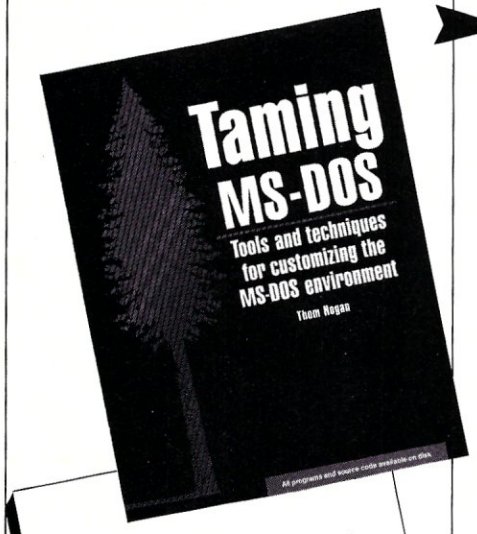
Nr is a text processor that is written in C, and is compatible with Unix Nroff. It includes complete implementation of the -MS macro package and an in-depth description of how -MS works.

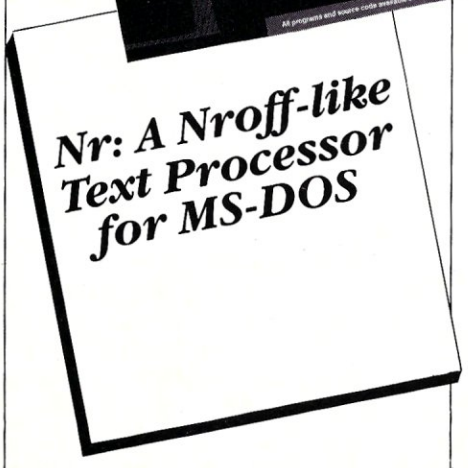
Nr will help you do hyphenation and simple proportional spacing. It supports automatic Table of Contents and Index, automatic footnotes and endnotes, italics, boldface, overstricking and understricking, and left and right margin adjustment. **Nr** also contains:

- extensive macro capability
- number registers in various formats including roman numerals and arabic, spelled out and in outline form.
- strings
- diversions and diversion traps
- input- and output-line traps

Nr comes configured for any Diablo-compatible printer, and Hewlett Packard's Thinkjet and Laserjet. It is easily configurable for most other printers and comes with full source code so that you can make it work with your system! Documentation is included. For PC compatibles.

Nr **Item #165** **$29.95**





To Order: Return this Order Form with your payment to M & T Publishing Inc., 501 Galveston Dr., Redwood City, CA 94063. *Or*, call **TOLL-FREE (800) 533-4372** Mon-Fri 8AM-5PM. In **CA call (800) 356-2002**

Item #	Description	Price

Subtotal_____

CA residents add applicable **sales tax** ____ %

Add $2.25 per item for **shipping**_____

TOTAL_____

Name_____

Address_____

City_____ State_____ Zip_____

☐ Check Enclosed.

Make Payable to M & T Publishing Inc.

Charge my ☐ VISA ☐ M/C ☐ Amer. Exp.

Card No._____ Exp.Date_____

Signature_____

33C

ate code in the appropriate memory model from the models that the 8086 segmented memory needs. Small model, intermediate model, large model, huge model—quick: which model does my program need? The small and middle models work reasonably well. Programs needing the large model may need significant hand-holding to port them to this system. I encountered some difficulty porting my large application package and others have reported similar problems with the large model. The standard libraries exist for each of the compiler models (small, middle, large)—that is, there is Slibc.a, Mlibc.a, and Llibc.a. The loader flags specify what version of the library to use. If you have your own application libraries and want to build programs with different models, you will probably have to build each library for each model. Don't blame SCO for this, though; blame Intel for designing this CPU and IBM for enabling it to survive.

The debugger is an antique, but again, don't fault SCO because its debugger is derived from the standard UNIX program adb. There is no comprehensive symbolic debugger like the sdb program called for in the System V Interface Definition. But there is adb, and most experienced UNIX programmers know how to use it. Symbolic debuggers and alternate compilers are available from third-party vendors.

One area that causes continual portability problems for application developers is the terminal interface. This system uses the System V standard termio system of system calls for controlling terminal setup (baud rate, parity, and so on). SCO XENIX V appears to conform to standards here.

NETWORKING

XENIX Network

The best supported network for XENIX systems is MICNET, presumably MicroSoft Net. It connects XENIX systems by serial (RS-232) cables and includes intermachine routing and forwarding. There is support for *rcp* (remote copy) patterned after the 4BSD command of the same name; a remote command facility patterned after 4BSD's *remsh* (or *rsh*); and electronic mail, about which more later. The information about routing, remote commands, and the like is stored in a few command files, and the network can be started up automatically when the machine is rebooted.

Old Faithful—uucp

Like every real UNIX, SCO XENIX V comes with the uucp communications program. The version shipped is not AT&T's latest (Basic Networking Utility, also called Honey DANBER after its authors). But when connected to an AT&T computer running the BNU version of uucp, it was able to transfer files without difficulty.

Electronic Mail

The main use of uucp links and many other networks on UNIX systems is electronic mail. SCO XENIX V comes with a version of the Berkeley Mail program (called mailx on System V, Release 2) for mail both on the same machine and to people on other machines. This mail system has an aliases facility based on MICNET and some XENIX-specific aliasing and forwarding utilities. Because they were different from the conventional System V mail and the Berkeley sendmail aliasing facilities, I decided merely to note the difference. Without having several XENIX boxes, I couldn't investigate the operation of MICNET or its forwarding. The system would probably work reasonably well for a collection of small XENIX systems in a local network connected by serial lines. And the mail interface is a standard one—versions of this mail program appear in both Berkeley UNIX and in System V.

DOS COMPATIBILITY

Because DOS is the normal operating system on PC-class machines, it should be no suprise that SCO XENIX has facilities for DOS users.

The hard disk can be partitioned into DOS and XENIX partitions. This is not easy to change once you've set it up, but it does allow you to have both XENIX and DOS programs living on the same disk.

But how do you access the DOS disk from XENIX? This version of XENIX has a set of routines for accessing the DOS disk. It provides access either to the DOS partition on a hard disk or a DOS floppy. These programs have names such as doscp, doscat and so on. They do the same to a DOS disk as their UNIX counterpart does—doscp copies files into or out of the DOS disk; dosrm removes a DOS file. You get the idea. What about dosdir and dosls? They give the same information: one gives you the report in the format a DOS user expects it, and the other gives it in a format that a UNIX or XENIX user would expect—not bad for compatability.

ADMINISTERING THE SYSTEM

Administering the UNIX system can be intimidating when you're asked to do it without proper background, just as driving in a stock car race would be intimidating if you hadn't learned to drive a car on the highway. Some UNIXs include a complete menu-driven system administration system that tries to guide the *naive superuser* (NSU)[3] through the process of running the system. One problem with this approach is that the NSU never learns what's going on underneath. SCO XENIX V doesn't have such a menu interface, but it does have shell files (command files) that automate a few of the more tedious tasks. Yet because they are shell files, you can easily inspect them to see what's going on. One of these adds new users: it puts the entry in the password file, creates the home directory and makes sure it's owned by the person, and handles the other routine tasks to create a new account. An apprentice superuser would want to start writing this program after a month or two of doing the tasks by hand; here it's given to you with the system.

There are many programs in service for doing backups on UNIX variants. Without discussing them in detail, suffice it to say that SCO XENIX V includes dump (and a clone called backup) and restore, as well as the perennial favorites tar and cpio, but it does not have System V's latest—finc/frec.

There are also several ways of accounting for UNIX disk usage. V7 UNIX had a fine program called quot for the purpose; to my mind it's superior to the system V acctdusg set of programs. SCO XENIX includes quot but not acctdusg. If you want the better program, it's here, but if you want pure System V, it's not.

One facility that SCO XENIX V offers that is often left out of low-cost UNIX boxes is the ability to reboot the system automatically without a human around. If the power goes off momentarily at 5 A.M. and nobody's around, it is handy if the system will restart itself before people show up at 7 or 8. This XENIX can sometimes reboot automatically; the system administrator can specify whether to reboot automatically on power-up based on whether the disks were checked before the system went down. Pressing any key on the console during an autoreboot converts the rest of the reboot to a manual mode.

UNIX keeps much of the hard-disk information in main storage until it really needs to be written (as a way of speeding up disk performance). If the system crashes or is shut down improperly, the disk may occasionally have a few files that are not updated correctly or other minor problems in the disk structures. A pro-

gram called fsck repairs 99.9999 percent of these problems automatically, and almost every UNIX variant runs fsck automatically when rebooted. SCO XENIX has a facility for checking whether the disks were stable when shut down and only running fsck if necessary. It does this by keeping a flag in the disk's master pointer list (the superblock); fsck clears the flag, and XENIX checks the flag when trying to mount or open the disk for the first time. Other systems do this checking differently or not at all. 4.2BSD, for example, uses a file called /fastboot to determine if the disks were checked before shutdown. Some older UNIXs just check the disk automatically every time you reboot. If you rebooted often, this would slow you down. SCO XENIX speeds this up.

In short, SCO XENIX V should be no harder to administer than a traditional V7 or Berkeley system.

SUPPORT

SCO XENIX is backed by The Santa Cruz Operation, a large player in the small-UNIX marketplace. Its other products include a Lotus 1-2-3 clone for XENIX and other add-on products.

Several levels of SoftCare support are available from SCO. Level 1 includes a newsletter and "prompt response to your written requests for software assistance." Level 2 includes a toll-free technical hot line and one free day of training. Each purchaser gets 30 days free hot-line support.

SUMMARY

SCO XENIX V offers most of the functionality of a full UNIX System on a PC/XT or PC/AT. It isn't a pure System V UNIX; there are bits of V7 and Berkeley UNIX and some XENIXisms scattered throughout. As a low-cost way to get started in UNIX, it should serve well. It looks like UNIX, and it feels like UNIX.

NOTES

[1] If you need a System V-like *file* command that works on V7 or Berkeley systems, look in the USENET group mod.sources around May 1987.

[2] Not counting public-domain programs such as the powerful Multiple Device Queueing System from the U.S. Navy Ballistic Research Labs.

[3] A term invented by some friends to describe the hordes of people who were being expected to run UNIX systems without any background in or training about UNIX. §

PRODUCT INFORMATION

SCO XENIX V, Release 2.0, for the 8088

Hardware requirements:
IBM PC/XT or reasonable clone; 640K of main storage; must have hard disk.

The Santa Cruz Operation
500 Chestnut St.
Santa Cruz, CA 95060
(408) 425-7222

Price:
Operating System—$595
Software Development System—$595
Text Processing System—$195
All three packages—$1295
Support—$145 per year (first 30 days free)

The SLR SuperLinker Plus is 3 - 10 times faster than any other linker, and look at these features:

- link a full 64K output (COM, HEX, SPR or PRL)
- works with Microsoft, Fortran, Basic, Cobol
- supports 32 character externals (SLR format)
- full drive/user support with alternate DU search
- supports 8 address spaces
- fill uninitialized spaces with 0 or FF
- global cross reference
- DSD80/SID compatible .SYM file
- manual overlays
- load map

$195

Requires: Z80 w/32 K TPA, CP/M 2:2 or greater

SLR Systems

1622 N. Main St., Butler, PA 16001
(800) 833-3061, (412) 282-0864
Telex 559215 SLR SYS

9-Track Tape Subsystem

for the IBM PC/XT/AT
XENIX or
MS-DOS.

The solution to your micro/mainframe communications problem is available today!

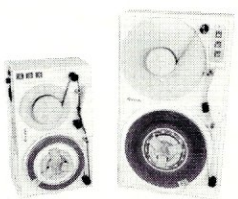
Qualstar's new ½ inch 9-track MINISTREAMER™ brings full ANSI data interchange capability to the PC. Now you can exchange data files with virtually any other computer using 9-track tape.

Available in both 7" and 10½" versions, the MINISTREAMER weighs in at only 27 pounds and uses less desk space than an ordinary sheet of paper, yet provides full 1600/3200 BPI capability at an affordable price. Up to 134 megabytes of data (depending on format) can be stored on a standard 10½" reel of tape, thus making the MINISTREAMER a highly-reliable answer to your backup requirements as well.

Tape subsystem includes tape drive, coupler card, cables, dust-cover and MS-DOS or XENIX compatible software. Prices start at $2,995.

386 READY!

Discover the many advantages 9-track tape has over other Micro/Mainframe links.



Call us today!

QUALSTAR®

9621 Irondale Avenue,
Chatsworth, CA 91311
Telephone: (818) 882-5822

Wendin's Operating Systems

by Stephen Randy Davis

Low-Cost VMS-like and UNIX-like Operating Systems for the PC—with Source Included

Readers of personal computing magazines will probably have noticed advertisements for one of Wendin's operating system products, be it the Operating System Toolbox, PCNX (formerly known as PCUnix) or PCVMS. Although these may seem unrelated, they are in fact interconnected. After realizing the similarities, I decided to review all three packages at one time to better describe each.

The most obvious similarity between the three is their low price tag—each of Wendin's products sells for $99. This makes Wendin's PCNX the lowest-priced UNIX on the market by roughly a four to one margin. This price includes a subscription to *The Wendin Kernel*, a newsletter of Wendin products and happenings. Wendin uses this newsletter to keep users informed of known bugs, coming upgrades, and new products. Product upgrades, when available, cost an extra $20. Technical questions can be referred to a non-800 number (8 to 5 PST). And, Wendin has a bulletin-board system (BBS) for additional user support and information exchange.

Each package comes in a largish 8½ × 11-inch softbound book of approximately 250 pages. A table of contents is provided in each, but a cross-reference is sorely missed. Within the pages of the book are inserted the license agreement and up to four floppies. All three packages include source code written in Microsoft C with interrupt handlers written in Microsoft Assembler. Although including the source was common in the early days of microcomputing, this practice is now rare. Because these systems are designed to be user tuned or, indeed, user rewritten, the presence of the source code in this case is almost a requirement and greatly appreciated.

THE OPERATING SYSTEM TOOLBOX

The Operating System Toolbox (OST) is also called the Personal Operating System Construction Set, which I think is an accurate description. The OST forms a DOS-compatible kernel that you can use to build any sort of multiuser, multitasking package desired. It is the OST upon which sit both the PCVMS and PCNX execs. This is the heart of the relationship between the three products.

Just like PCBIOS.SYS, you do not run the OST as a normal program. For the OST to do anything, you must write a shell program. An example shell is provided, but it does little more than ask your name and give it back to you. The documentation provided with the OST explains in detail how to go about writing a shell, but, like Heathkit says, this kit is not for beginners. Although a shell could be written in any language, support in the form of library routines is provided specifically for the Turbo Pascal and Microsoft C languages.

Let me explain shells briefly using DOS as an example. The PC-DOS operating system actually consists of three files: PCBIOS.SYS, PCDOS.SYS, and COMMAND.COM. The first two files are permanently loaded at boot up. The Interrupt 21 DOS system calls made by your application software (refer to William Wong's articles in *Micro/Systems Journal,* May/June 1985 through January/February 1986) are handled by these permanently resident files. COMMAND.COM is not permanently resident, and your software cannot reference it. Similar to one of your programs, COMMAND.COM runs on top of DOS. Its job is to read keyboard input and to try and make sense of what is entered. When you enter ERASE, it is COMMAND.COM that decides you are trying to delete a file. It then uses DOS calls to delete whatever file is indicated. When you enter the name of your favorite application program, COMMAND.COM finds it on the disk, loads it, and passes control to it—all using DOS system calls.

The OST corresponds to the PCBIOS-.SYS and PCDOS.SYS. PCNX and PCVMS sit on top of the OST, reading keyboard input using the system calls defined by the OST. The OST does not need to rely on DOS, except that the current version (1.0) uses Interrupt 21 type calls for file access type calls. Therefore, DOS as well as the OST must be RAM-resident. Although this uses up more memory, it does ensure DOS file compatibility. The OST's system calls are patterned after those of DEC's VAX VMS operating system, Version 2.0. Some 80 to 90 VMS system calls appear in the OST. Like VMS and unlike DOS, the OST is multitasking and multiuser.

The OST filters DOS system calls into their VMS equivalents to provide DOS software compatibility. Unfortunately, this compatibility is not complete. The OST has no overlay loader and does not support undocumented DOS calls. In addition, the OST cannot properly support programs that steal (as opposed to borrow) interrupts. Multiuser capability is provided via terminals attached to the PC's serial ports.

EVEN MORE POWER AND FLEXIBILITY

BRIEF 2.0

Users and industry press alike have unanimously proclaimed BRIEF as the best program editor available today. Now, the best gets better, with the release of BRIEF 2.0.

Straight from the box, BRIEF offers an exceptional range of features. Many users find that BRIEF is the only editor they'll ever need, with features like real, multi-level Undo, flexible windowing and unlimited file size. But BRIEF has tremendous hidden power in its exclusive macro language. With it, you can turn BRIEF into your own custom editor containing the commands and features you desire. It's fast and easy.

Jerry Pournelle, columnist for BYTE magazine summed it all up by saying BRIEF is, "**Recommended. If you need a general purpose PC programming editor, look no further.**" His point of view has been affirmed by rave reviews in C JOURNAL, COMPUTER LANGUAGE, DR. DOBB'S JOURNAL, DATA BASED ADVISOR, INFOWORLD AND PC MAGAZINE.

One user stated "BRIEF is one of the few pieces of software that I would dare call a masterpiece." Order BRIEF now and find out why. **BRIEF 2.0 is just $195. If you already own BRIEF, call for upgrade information.**

TO ORDER CALL: 1-800-821-2492 (in MA call 617-659-1571)

As always, BRIEF comes with a 30 day money-back satisfaction guarantee.

Solution Systems™
335 Washington St.
Norwell, MA 02061
(617) 659-1571

Look at these BRIEF 2.0 enhancements!

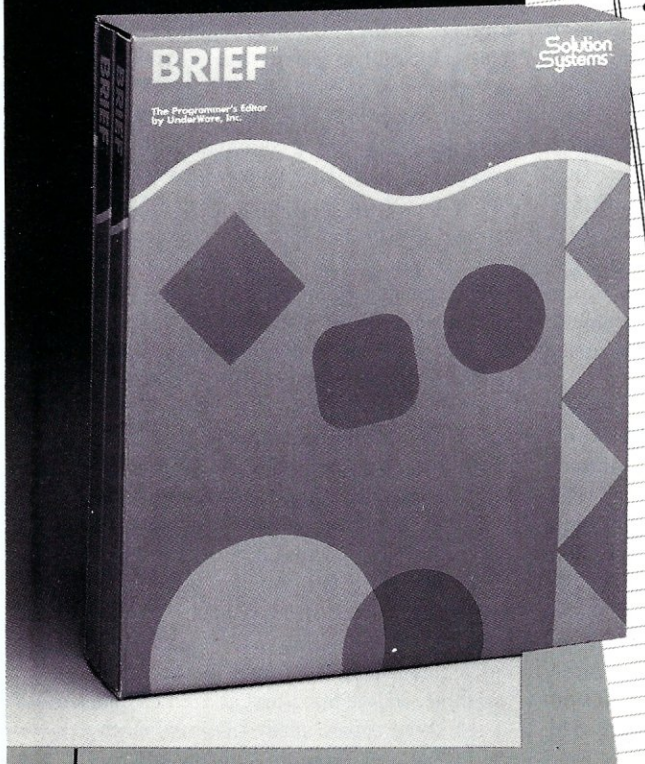
Main Features:

- *All new documentation with tutorials on basic editing, regular expressions* ***and*** *the BRIEF Macro Language.*
- *Setup program for easy installation and configuration. (Requires no knowledge of the macro language)*
- *Increased speed for sophisticated operations like Undo and Regular Expression Search.*
- *Expanded regular expressions, with matching over line boundaries.*
- *More block types, with marking by character, line or column.*
- *Command line editing (move cursor, add and delete characters, specify command parameters).*
- *Support for more programming languages.*
- *Optional borderless windows.*
- *Enhanced large display support, including wider displays.*
- *Reconfigurable indenting for C files (supports most indenting styles).*

Plus the basic features that made BRIEF **SO** *popular!*

Basic Features:

- *Full multi-level Undo*
- *Windows*
- *Edit many files at once*
- *File size limited only by disk space*
- *Automatic language sensitive indentation*



Requires an IBM PC or compatible with at least 192K RAM.

BRIEF is a trademark of UnderWare, Inc.

Solution Systems is a trademark of Solution Systems.

Programs that write directly to the system screen can run perfectly on the system console but do not run properly from the serial ports. Within these limitations, shells written using the OST kernel, including PCNX and PCVMS, should be able to execute "well-behaved" DOS programs.

Note that a shell need not be anything like COMMAND.COM, PCNX, or PCVMS. If I were writing a spreadsheet, for example, I might want to make it multitasking so that users could continue to enter data during those long recalcs. I could start with the OST, the spreadsheet program being my "shell." I might do the same if I were designing a database to allow multiple, simultaneous users. Because the OST supports DOS system calls, I would be free to write my application either as a DOS application or a VMS application, as long as I remembered the DOS incompatibilities noted earlier. Wendin allows products developed with the OST to be sold without royalties (the source and object code are licensed solely to the purchaser, however).

As of this writing, Wendin is readying Version 2.0 of the OST for release. The biggest improvement in this new version is that it will have its own file system and thus will not require DOS to be RAM-resident. This should increase performance as well as provide support for different shell path names (most important, DOS' \, UNIX's /, VMS' [], and Tops' username). Version 2.0 will continue to support DOS formatted disks and should address some of the other compatibility problems outlined earlier.

UNIX—PCNX

Although source code is provided for the PCNX shell, the source for the OST kernel, upon which PCNX relies, is not included. As with any OST application, PCNX provides support for one user on the system console and up to two more users via the RS-232 serial ports. PCNX requires at least 256K of random access memory. Only a minimal PCNX can be installed on 360K floppies because the entire PCNX system consists of 900K of executable files.

Users must log in to PCNX. The system comes configured with only the master user, root. Root can add other users and assign them a password and access privileges via the *mkuser* command. Passwords of any length, including null passwords, are allowed. Users can log in from either the user console or from serial ports once these have been included in the system with the *getty* command. To return to DOS from PCNX, a privileged user enters the username SHUTDOWN at the login prompt. Although this technique is unique to PCNX, it is pretty intuitive. Upon "shutting down," the user is returned to DOS.

PCNX has much the same feel and appearance as UNIX. It does not hurt that the underlying file system used by DOS is very similar to that used by UNIX. Wendin did a good job of including all the standard commands from the Bourne shell. Commands were also added from other UNIX versions, such as 4.2BSD. The differences I was able to find between PCNX and "real UNIX" were small—for example, PCNX does not maintain UNIX-style file access flags and does not distinguish between lowercase and uppercase as UNIX does. Anyone using PCNX to learn UNIX commands would have little trouble migrating to other UNIX systems.

PCNX has little support for programs written for true UNIX environments, however. Because the OST has more of a VMS structure, the underlying system calls have nothing to do with normal UNIX system calls. Of course, PCNX does support DOS system calls. Because the standard function library that comes with most MS-DOS C compilers is based upon the UNIX system, this provides some support for C programs written for the UNIX environment. Still, PCNX qualifies as UNIX only at the user level.

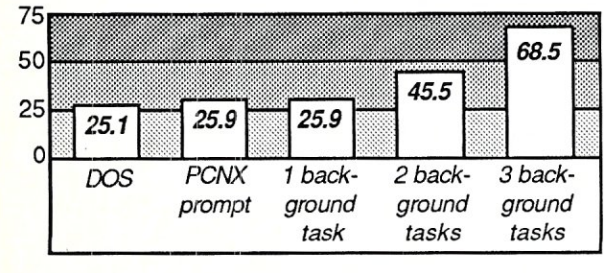


Table 1. Assembly of three different copies of the same source file using Microsoft MASM assembler. Three background tasks represent the maximum number of tasks that could be loaded into a 640K PC.

In order to speed up response time, the most common commands in DOS, such as *DIR* and *DEL*, are built into COMMAND.COM. These so-called resident commands respond immediately when they are invoked. There are, however, no resident commands under PCVMS; every command must be fetched from a disk. On a hard-disk system, this presents little problem, but on a floppy system, this is a real nuisance because it means that the PCNX system disk must remain in drive A at all times.

Several PCNX commands are of particular note. When the makers of DOS defined the format of the .EXE executable file, they included a field in the header specifying the amount of memory required by the program. Unfortunately, most Microsoft software tools do not set it. Therefore, applications built with such tools specify in their headers that zero memory is required. Under single-user, single-tasking DOS, this is no problem because all of memory is available anyway; however, this does present a problem in multitasking or multiuser systems in which several applications must share available memory. PCNX solves the problem by providing a utility, exefix, with which the user can change the memory requirements specified by the .EXE file. Of course, the user must guess at the proper amount of memory an application needs, but it's better than nothing.

Other commands of particular note are those dealing with multitasking. Under PCNX, the user is not required to wait for the completion of a command, as with DOS. If the user simply appends an ampersand (&) to a command, PCNX executes the command as a separate job, returning the user a prompt immediately. This can result in greatly increased work throughput—for example, editing of one file can proceed while another is being compiled. Spawning a command with the ampersand causes the background job to run with the same priority as the foreground. Jobs of the same priority share the system equally. The commands *nice* and *nohup* can be used to run background jobs at a lower priority, which causes them to use less system time than the foreground, so as to not adversely affect foreground performance. The status of the currently active jobs can be examined with the *ps* command, and active jobs can be prematurely terminated using the *kill* command.

The 8088 microprocessor is not particularly suited to multitasking, which results in disappointing performance with some multitasking operating systems. To test PCNX's multitasking performance, I made three identical copies of the same assembler source file. I first assembled one of the three copies under DOS and from the PCNX user prompt using the Microsoft MASM assembler. I then assembled

one, two, and three copies simultaneously in the background. The results of this test appear in Table 1. In each case, time was measured from the first assembler preamble message until the first "NO ERROR" message. It is interesting that the times for DOS prompt, PCNX prompt, and single background job were roughly the same. This indicates that PCNX is just about as fast as DOS is and that it takes little time to start another job. Notice also that the times for two and three assemblies were not two and three times as long as for one assembly. This is because of certain efficiencies gained from multitasking. PCNX did feel slightly slower primarily because of the absence of resident commands, but overall I would rate PCNX's multitasking performance as good.

I then tested PCNX compatibility by running dozens of DOS programs from my hard disk. PCNX had some problems running several DOS programs, in addition to those mentioned in connection with the OST. First, control characters such as tabs were not interpreted on output. Rather than white space, a tab generated an o. In addition, some programs appeared not to see keys that generated scan codes—this included arrow and function keys. This may have had something to do with the necessity for supporting ANSI terminals, which do not behave identically to the PC's screen and keyboard. Either way, the screen and keyboard control programs appeared to need some work.

PCNX simply did not support a lot of programs on my disk. True, it would be asking too much for PCNX to support debuggers and intricate file utilities because these are intricately interwoven with the fabric of DOS itself. There were some mundane programs, however, that had me reaching for the big red switch under PCNX. To its credit, all the program development type applications I tested ran without a hitch. These included Microsoft's assembler, various C compilers, Turbo Pascal, the DOS linker, and the like.

Not too surprisingly, I encountered problems with terminate and stay resident (TSR) programs. If executed before entering PCNX, they either worked properly, had no effect, or crashed; there was no way of predicting the outcome. Attempting to install a TSR program from PCNX was disastrous because it always caused the user's process to suspend. Wendin claims that spawning a TSR program with the ampersand generally works properly, but the several TSR programs I tried did not seem to work. I don't think my hard disk is typical, being cluttered with "tricky" programs, but I think DOS compatibility could be improved. Hopefully the spectrum of programs PCNX will support will grow under Version 2.0.

PCVMS

PCVMS has much the same feel as PCNX, with most of the observations concerning PCNX applying equally well. This is not surprising because both are based on the same OST. Because of the great similarity, I will highlight only those features unique to PCVMS.

The commands used under PCVMS are virtually identical to those used under VAX VMS. UNIX had not been so different from DOS, so it was eerie to run DOS programs using VMS commands. PCVMS supports VMS-style logical device names, both on output and on input—for example, the disk upon which the system boots can be referred to as SYS$BOOT rather than C:. Because Version 1.0 of the kernel uses the DOS file system, however, path names appear in DOS style and not VMS style. In addition, PCVMS does not support version numbers—when a file is over-

• Z Best Sellers •

Z80 Turbo Modula-2 (1 disk) $69.95
The best high-level language development system for your Z80-compatible computer. Created by Borland International. High performance, with many advanced features; includes editor, compiler, linker, 552 page manual, and more.

Z-COM (5 disks) $119.00
Easy auto-installation complete Z-System for virtually any Z80 computer presently running CP/M 2.2. In minutes you can be running ZCPR3 and ZRDOS on your machine, enjoying the vast benefits. Includes 80+ utility programs and **ZCPR3: The Manual**.

Z-Tools (4 disks) $150.00
A bundle of software tools individually priced at $260 total. Includes the ZAS Macro Assembler, ZDM debuggers, REVAS4 disassembler, and ITOZ/ZTOI source code converters. HD64180 support.

PUBLIC ZRDOS (1 disk) $59.50
If you have acquired ZCPR3 for your Z80-compatible system and want to upgrade to full Z-System, all you need is ZRDOS. ZRDOS features elimination of control-C after disk change, public directories, faster execution than CP/M, archive status for easy backup, and more!

DSD (1 disk) $129.95
The premier debugger for your 8080, Z80, or HD64180 systems. Full screen, with windows for RAM, code listing, registers, and stack. We feature ZCPR3 versions of this professional debugger.

Quick Task (3 disks) $249.00
Z80/HD64180 multitasking realtime executive for embedded computer applications. Full source code, no run time fees, site license for development. Comparable to systems from $2000 to $40,000! Request our free Q-T Demonstration Program.

Z-System OEM inquiries invited.
Visa/Mastercard accepted. Add $4.00 shipping/handling in North America, actual cost elsewhere. Specify disk format.

Echelon, Inc.
885 N. San Antonio Road • Los Altos, CA 94022
415/948-3820 (Order line and tech support) Telex 4931646

written, it is gone, unlike on the VAX.

Device names are similar to those of VAX VMS, such as TTA0 for the console, TTA1 for COM1, and TTA2 for COM2. The disk drives carry the default names DRA0 through DRA5. These names can be changed. Certain logical device names have particular significance, such as SYS$BOOT, SYS$ERROR, SYS$INPUT, and SYS$OUTPUT. These can be redirected to different devices, if desired. For example, SYS$ERROR might be directed to the printer, forcing error messages to be logged there. Unlike UNIX, PCNX, and DOS, only three path directories can be searched for user files; these are designated PATH1, PATH2, and PATH3.

As with PCNX, PCVMS users must log in before use. VAX VMS requires passwords of at least six characters whereas PCVMS allows any length password, but otherwise the login process is the same. Only the first three letters of commands are parsed under PCVMS—thus, *DIR* is the same as *DIRECTORY* or *DIREKTORIE*. This is unlike VMS, which only requires that the command entered be unique but scans whatever is present. Also like PCNX, entering *SHUTDOWN* at the login prompt returns the user to DOS.

Multitasking is handled under PCVMS with the *SPAWN, SUSPEND, RESUME, STOP,* and *DPD* commands. Entering *SPAWN* followed by an executable file name causes that program to run as a new process, returning a prompt immediately. *SUSPEND, RESUME,* and *STOP* allow operator control of already running processes. *DPD* displays the status of a selected process every half second on the user's screen. The *DPD* display itself is something of a lesson in multitasking execs. *SHOW SYSTEM* displays all the processes currently running in the system. Multitasking performance of PCVMS is similar to that of PCNX.

CONCLUSION

Wendin's Operating System Toolbox and related PCVMS and PCNX are difficult to evaluate in just a few lines. MIS managers who want to establish a multiuser VMS or UNIX system on their PCs from which they can run all their favorite management DOS software, such as Lotus 1-2-3, dBASE, Framework, and so on, will not find their answer in PCNX or PCVMS.

For hackers who would like to learn the ins and outs of multitasking, multiuser operating systems, I would heartily recommend the OST and either PCVMS or PCNX for its source code and documentation. Curious users interested in teaching themselves the UNIX operating system will have no trouble migrating from PCNX to any other UNIX machine. The same could be said of PCVMS, despite its slight peculiarities. Similarly, mainframe users who are accustomed to UNIX or VMS at their place of work and who are not interested in mastering DOS on their machine at home, might find PCNX or PCVMS more to their liking. That is, provided their machines can properly support the DOS software they intend to run.

Overall, I was left with a good feeling about Wendin's operating systems, but I can't say why. Maybe it was the low price, maybe it was the helpful technical support, or maybe it was the the idea of a small company with such lofty goals. I found enough DOS software that the products wouldn't execute properly that I was left with a bit of an uneasy feeling, though. I hold out great hopes of increased compatibility for Version 2.0, but even as it is, I find all three packages worth the modest investment. §

Stephen Randy Davis is a senior systems programmer for a defense contractor in Greenville, Texas, where he programs various microprocessors. He is also working on his Masters in physics.

PRODUCT INFORMATION

Operating System Toolbox (OST)
An MS-DOS-compatible kernel for building a multiuser, multitasking operating system. $99

PCVMS
A DEC VMS-like shell that works with OST to form a VMS-like system. Source code included. $99

PCNX
A UNIX-like shell that works with OST to form a complete UNIX-like system. Source code included. $99

All products include subscriptions to the Wendin newsletter, access to the Wendin BBS, telephone support, and upgrades ($20 each).

Wendin Inc.
P.O. Box 266
Cheney, WA 99004
(509) 235-8088

Introducing Periscope™ III

A new generation of debugging for the IBM PC, XT, AT and close compatibles

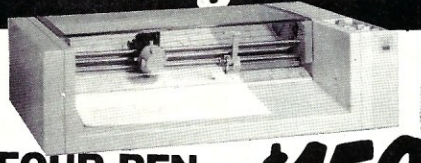
Now you can invest $995 and get the most powerful debugging tool available short of a $10,000 in-circuit emulator! The Periscope III board's hardware breakpoints and real-time trace buffer help you solve the really tough debugging problems. If you ever deal with errors in real-time systems, intermittent failures, interfacing with undocumented systems, or bottlenecks in your code, Periscope III may be just what you need!

Call TOLL-FREE 800/722-7006 for more information.

The PERISCOPE Company, Inc.
14 Bonnie Lane • Atlanta, GA 30328 • 404/256-3860

California Digital

17700 Figueroa Street • Carson, California 90248



FOUR PEN COLOR PLOTTER $159

The manufacturer has asked us not to publish their name. But this four color plotter was produced by one of the Worlds largest makers of computers.
The 410 color plotter will connect to the **serial port** of virtually any microcomputer. Simple ASCII commands direct one of the four color pens to draw circles, arcs or ellipses on paper or transparency material up to 11 by 17 inches. The plotter is capable of producing the full upper and lower case alphabet along with seven international character sets. Text can be printed horizotal, vertical or diagonal in sizes from 1/16 to 6 inches.

Enlargements or reductions are achieved through elaborate firmware. Pen travel is four inches per second with .004" pen resolution. Standard pens are available in an assortment of 32 different colors and widths.

The ideal plotter for architecture, CAD engineering or graphic design. **At $595 it was a great buy, at $159 its a steal.** Support packages for specific computers available. Manual only $15 refundable upon purchase of plotter.

OTHER PLOTTERS AVAILABLE:
Hewlett Packard • Houston Instruments • Roland Sweet P • CalComp... please call for prices.

$2495

80 Character Daisy Wheel Printer $759

These Fujitsu Daisy Max 830 were manufactured for Motorola's Computer Division. The purchase order was canceled and Fujitsu was forced to liquidate these 80 character per second daisy wheel printers at "fire sale" prices.

Features: bullet proof construction, serial RS-232 interface, Diablo 630 wheels and commands, programmable line spacing in increments of 1/96" and column spacing of 1/120". The printer is also capable of underscoring, bold overprint, shadow print, centers and justifies along with vector plotting.

Factory suggested price of the Daisy Max 830 was $2495, while supplies last California Digital is offering this liquidated special at only $759. Also available: tractor and sheet feeders.

COLOR GRAPHICS TERMINAL $659

~~$3995~~

The Lear Siegler 7107 color graphics terminal is both DEC (VT-100 / VT-51) and Tektronix (4010/4014) compatible.
The terminal offers a palette of 4,096 colors with a resolution of 640 by 480 pixels on a 13 inch non-glare black screen.
Horizontal pan and image zoom, to 16 times original size, are standard features of this color graphics terminal.
The 7107 is the ideal terminal for CAD/CAM applications.

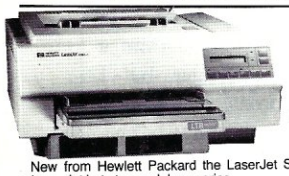
NOW YOUR COMPUTER CAN READ!!

Omni-Reader... the first optical character reader **designed and priced for the small computer** $179

COPY:
•Manuscripts •Contracts •Articles •Forms •Invoices

APPLICATIONS:
•Mailing Lists •Editing •Data Base Management •Transfering information between incompatible systems

Uses a standard RS-232 serial port hookup to interface eaisily with your computer.



HEWLETT PACKARD

LASER JET II $1895

New from Hewlett Packard the LaserJet Series II. More features than the LaserJet but at a much lower price.
Standard on the Series II is 512 K/byte memory expandable to 4 Megabytes, both serial and parallel ports. List price $2495 California Digital price $1895. Memory upgrade, type fonts and toner cartridges available.

Other Laser Printers available:
Texas Instruments • Okidata • Apple • Ricoh Qume • AST Research • QMS • Xerox • Canon Quadram • Centronics • CIE C.Itoh • DataProducts

Bernoulli Box $1595

was $3540 now only

The **Bernoulli Box** by Iomega, features 10 and 20 megabyte removable cartridges, and delivers reliability, expandability, transportability, security and speed in one versatile subsystem. It lets you transfer megabytes of information safely and swiftly for primary or backup storage. Or combine several software programs onto a single cartridge for easy switching from one to another.

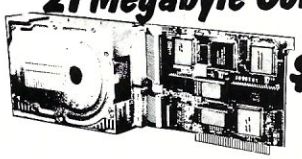
Reliable... the Box has incredible resistance to shock and vibration completely eliminating the possility of head crash.

Expandable... grow at your own pace by adding inexpensive cartridges. When **security** is essential, don't lock up your system... just lock up the cartridges.

The Bernoulli Box delivers preformance that often exceeds the best of hard disk speed and the convenience of floppy disks. At these prices don't be caught wishing you had one after a loss of irreplacable data.

		List	Our Price
10+10 Meg.	A2210H	$3450	1595
20+20 Meg.	A2220H	4540	2095
Bootable Controller		255	189
10 Meg. Cartridge		79	49
20 Meg. Cartridge		99	65

21 Megabyte Gold Card $479



Also available 31 Megabytes at only $559.

The fastest, lowest powered, longest warrantied, most durable, highest capacity, most reliable, lowest priced HARD-DISK-ON-A CARD available in the world today.

• 65ms Access Time • *Fastest Available* • Automatic Head Unloading • *Protects Heads and Media* • 2K Sector Buffer • *Increases System Throughput* • High Reliability: 28,000 hr. MTBF *No one else even close* • 15.5 Watt Power Consumption • *Lowest Available* • 2 Year Warranty *Longest Available*

$35

We have them again... The Remex 480 was a sell out two years ago but we located an additional 10,000 units and are offering these 2/3 height IBM/PC compatible 360K/Byte drives at only $35 each at quantity two.

85

The Mitsubishi 4851 was probably the best 360 K/Byte drive ever produced. The production cost was too high and Mitsubishi discontinued the product. Quantity two $85 each.

115

The Panasonic 475 is jumper selectable between 96 TPI format and 1.2 Megabyte as used in the IBM/AT computer.
Suggested price of the 475 is $189 but California Digital located a computer manufacturer that had to liquidate stock making available 1000 of these choice drives. Quantity two $115 each.

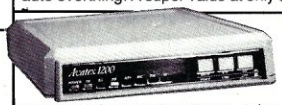
	One	Two	Ten
REMEX 480 ⅔ height for PC	39	35	call
MITSUBISHI 4851 360K/Byte	89	85	82
MITSUBISHI 4853 96 TPI.	99	95	92
MITSUBISHI new 501 half ht.	129	119	109
MITSUBISHI 504A AT comp.	169	159	155
TEAC FD55BV half height	109	99	89
TEAC FD55FV 96 TPI, half ht.	119	109	105
TEAC FD55GF for IBM AT	169	159	155
PANASONIC 455 Half Height	109	99	89
PANASONIC 475 1.2 Meg./96	119	115	109
Switching power supply			49
Dual enclosure for 5¼" drives			59

Smarteam 2400 Baud $289

The Smarteam 2400 offers all the features of the Hayes Smart Modem 2400 for a fraction of the price. Now is your opportunity to purchase a 2400 baud modem for only $289.
Also available: The Smarteam 1200 at only $139

U.S. Robotics 2400 Internal $189

The US Robotics Micro 2400 modem is one hundred percent Hayes compatible, auto dial, auto answer, auto everthing. A super value at only $189.

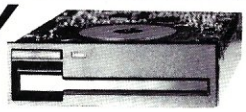


$99

This 300/1200 baud modem matches design specs of Bell 212A and for all practical purpose those of the Hayes Smart Modem. The Avatex brings you reliable data transmission for only $99.

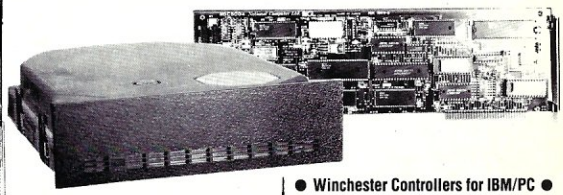
OTHER MODEMS AVAILABLE:
Hayes • Universal Data • Fujitsu
Prometheus • Anchor • US Robotics

SONY $139



The Sony 53W is a 3½ inch double sided double density disk drive. The drive can be connected to your existing floppy controller but will require MS/DOS 3.2 to properly operate in the PC and compatibles. The 53W is similar to the drive being used in the PC portable.

Seagate 30 MEGABYTE WINCHESTER HARD DISK KIT $459



Five Inch Winchester Disk Drives

	each	two+
SEAGATE 225 20 Meg. ½ Ht.	319	299
SEAGATE 238 30 Meg. RLL	369	359
SEAGATE 4026 26 M. 35mS.	659	629
SEAGATE 4051 51 M. 35mS.	759	739
FUJITSU 2242 55 M. 35mS.	1399	1329
FUJITSU 2243 86 M. 35mS.	1895	1819
RODIME RO-202E 27 Meg.	659	629
RODIME RO-203E 40 Meg.	695	659
RODIME RO-204E 53 Meg.	995	959
CONTROL DATA 94155-86 M	1829	1779
MAXTOR XT1140 140 Meg.	2595	2529
TOSHIBA MK56 70 M. 30mS.	1789	1729
TANDON 502 10 Meg.	419	379

• Winchester Controllers for IBM/PC •	
XEBEC 1220 with floppy controller	229
DTC 5150CX	139
OMTI 5520 half card	129
OMTI 5527 RLL controller	159
ADAPTEC 2070 RLL controller	179
ADAPTEC 2010A	159
WESTERN DIGITAL WD/1002WX2	119
• SCSI/SASI Winchester Controllers •	
XEBEC 1410A 5¼" foot print	219
WESTERN DIGITAL 1002-05E 5¼"	289
OMTI 20L	119
• Winchester Accessories •	
Installation Kit with manual	10
Winchester enclosure and supply	139
Dual 20/34 cable set	25
Switching power supply	49

VISA

master charge

Shipping: First five pounds $3.00, each additional pound $.50.
Foreign orders: 10% shipping, excess will be refunded.
California residents add 6½% sales tax. • COD's discouraged.
Open accounts extended to state supported educational institutions and companies with a strong "Dun & Bradstreet" rating.

TeleFax
(213) 217-1951

TOLL FREE ORDER LINE
(800) 421-5041
TECHNICAL & CALIFORNIA
(213) 217-0500

VENIX System V

by Jim Rosenberg

A UNIX System V, Release 2, Implementation—Recommended for Some but Not All

VENIX System V, by VenturCom, is a fully licensed port of UNIX System V, Release 2, and is available for IBM PC/XTs, PC/ATs, and close compatibles as well as certain other similar machines. It is sold by Unisource Software and also by VenturCom. (Originally only Unisource sold VENIX.) VenturCom's documentation only guarantees that it will work on genuine XT and AT hardware, but Unisource supports a variety of clones and disk controllers.

I evaluated VENIX on an AT&T 6300 using a DTC 5150BX controller and a 32MB Tulin 640 hard-disk drive. It requires at least 512K, though it will probably boot on a 256K system, and as with any serious version of UNIX, a hard disk is needed for anything practical. The entire VENIX V package requires about 9MB of disk space, though you may not need all the programs that come with it.

I've been using VENIX 1.18 and later 2.0 for more than two years, so many of my comments contrast VENIX V with the earlier versions as well as with The Santa Cruz Operation's XENIX V (SCO V). Unisource declined to provide a System V upgrade as a review copy. Instead, they agreed to exchange the System V upgrade for my copy of the Connector, a program that runs DOS and UNIX concurrently. I believe the Connector is probably healthy on fully IBM-compatible hardware, but it would not run correctly on a 6300 and probably has serious compatibility problems with some disk controllers.

VENIX V is a complete UNIX System V with few things missing and has a nearly complete set of Berkeley enhancements. Among the pieces missing are the symbolic debugger sdb, troff, ratfor, and the Berkeley programs tset and strings. Fortunately, strings is available on Unisource's dial-in machine. One astonishing omission is the lack of any mention in the manual of the *acct( )* system call. This establishes process accounting, which makes a log entry for each process that UNIX runs. VENIX does have process accounting, though the utility to print out the logs is missing. Undocumented system calls may be routine in the DOS world, but they're unheard of in the UNIX world!

FUNCTIONALITY

The overall performance of VENIX V is excellent, given the limitations of an 8-MHz 8086 and a 90-msec hard disk. I have to snort out loud every time I hear someone pontificate that you need a 80286-, or better yet, an 80386-based system to run multitasking on a microcomputer.

No amount of magic will turn a PC compatible into a VAX. VENIX would be more than suitable on a PC as a single-user work station or as a multiuser system where most of the time only one user is doing highly disk-intensive work at one time. Three concurrent users all trying to compile C simultaneously will find it intolerable, and two will find it uncomfortable.

I've resisted the temptation to publish the standard *Byte* benchmarks because I believe that the subject of benchmarking UNIX is so complex that these do not prove much and figures for a 6300 will mean little to PC or AT users. The feel of the response time seems quite reasonable for low-end hardware.

If VENIX has one performance weakness, it's the length of time it takes to wake up a process that's been swapped. I left a copy of vi (the Berkeley UNIX screen editor) idle for 24 hours and then pressed ^G to display a status line. It took nearly 10 seconds! Although my hard disk is on the slow side, this could hardly account for it because, after the initial seek, swapping is not very seek intensive—swap space is a set of contiguous disk blocks.

The stability of the kernel has been almost perfect in three weeks of a typical mix of word processing and C software development. I did get one system crash from a breakpoint in adb, the debugger, but I haven't been able to recreate it. I have a high regard for VenturCom's ability to produce an exceptionally stable kernel.

VenturCom was first out the door with a UNIX that supports virtual terminals on an XT. Other versions of UNIX that support virtual terminals are SCO V and Microport System V for the AT. VENIX provides four virtual terminals, and to use this feature you need a Color Graphics Adapter or compatible video card. The advantage of this approach is that the four screens do not detract from the amount of RAM available to user processes.

SCO V allows six virtual terminals on PC compatibles and ten on ATs. Each virtual terminal acts as a totally separate login; by pressing Alt-1 - Alt-4 (Alt-F1 - Alt-F10 on SCO V), you toggle among the various virtual terminals. This is so incredibly useful I always feel like my machine has terminal arthritis when I have to boot DOS. For instance, you could have the debugger on one virtual terminal, a C source file on another window, a C .h include file on a third window, and one still left over for a shell. Note that each window is full-screen, though—you see all or nothing, with no overlap.

VENIX V fixes the most hideous weakness of earlier versions of VENIX. The console tty driver for 1.18 and 2.0 called the BIOS. The performance was so bad that the UNIX communications program cu would drop characters at 300 baud with concurrent intensive screen use by another process! The new console driver writes directly to screen RAM, and cu now happily keeps up at 1,200 baud. I've even been

For the professional who doesn't want or need protection from the DOS prompt

The COMMAND System™

The proper way to organize files on a hard disk is in a deeply nested tree structure. But most of us don't do it that way because it's too much trouble to refer to the directories at the other end of the tree. The COMMAND System gives you the way to do it.

By giving each subdirectory a short name (or letting one of our programs do it for you), you can access every file on your disk easily, just like you can in many more powerful operating systems.

Instead of typing C:\WRITING\DOCUMENT\COMMAND\MAGIC\
You can type just MAGIC\\

- Instant Access to Any Directory on Your Hard Disk
- Automatically Generated Short Names For Every Directory
- All These Programs Use Short Names
- Fast Programs to Copy, Erase, Rename, Move, Compare
    - Consistent Command Structure
    - Multiple Operations On a Line
    - Built-in Help
    - Super Programs to Show Files and Directories
- CompuMagic's SEARCH Program — Search ASCII or Word Processor Files by Line or Paragraph
- Uses Less than 20K of Memory

SPECIAL INTRODUCTORY OFFER EXTENDED

Save $20 from the regular $95 price.
Order before June 12th and pay only $75.

30-day Money-Back Guarantee — Never Copy-protected

The COMMAND System requires DOS 2.0 or higher and includes Full Documentation and a disk with 25 programs. Call Now with your COD, VISA or MC order, or send check or MO. Free UPS Ground Shipping. Add $2 for COD, $5 for overseas. (MD res. add 5%)

We still support CP/M too!
The CompuMagic Utility Package $45 — SEARCH $30

CompuMagic, Inc.™

P.O. Box 437
Severn, MD 21144
(301) 969-8068

MAKING COMPUTERS WORK MAGIC SINCE 1983

able to run cu at 9,600 baud successfully, though at 9,600 baud vi occasionally mangles screens.

As a simple example of VENIX V's conformity to the AT&T standard, I tried compiling C-Kermit using the off-the-shelf makefile entry for System III/System V. It compiled without a hitch and worked beautifully. There are a few differences in system calls between VENIX V and the standard, but they will affect few programmers. The major difference concerns shared memory. VenturCom has gone off on its own here, but my feeling is that few UNIX programmers are using this.

VENIX allows negative numbers for alarms and sleeps, which indicate that the time should be measured in clock ticks rather than seconds. VENIX V has implemented the /usr/group record and file locking scheme, which may be different from AT&T's scheme in System V, Release 3. XENIX uses its own scheme of locking and allows share locking, in which any number of readers are allowed but no writers. Omission of share locking is a weakness in the /usr/group standard.

A strange omission from standard UNIX practice, which you can remedy yourself if you know how, is VENIX's lack of a program called update, which to oversimplify a bit, causes the UNIX file system buffer cache to be flushed every 30 seconds. Instead VENIX has a scheme called autosync for flushing on exit from programs. I've never understood the reasoning and wrote my own update daemon.

THE VENIX C COMPILER

Obviously the C compiler is a pretty important cornerstone in any version of UNIX, so it should be evaluated carefully. I have some bad news here. VenturCom has no large-model C compiler, and there isn't even any indication that one will be forthcoming.

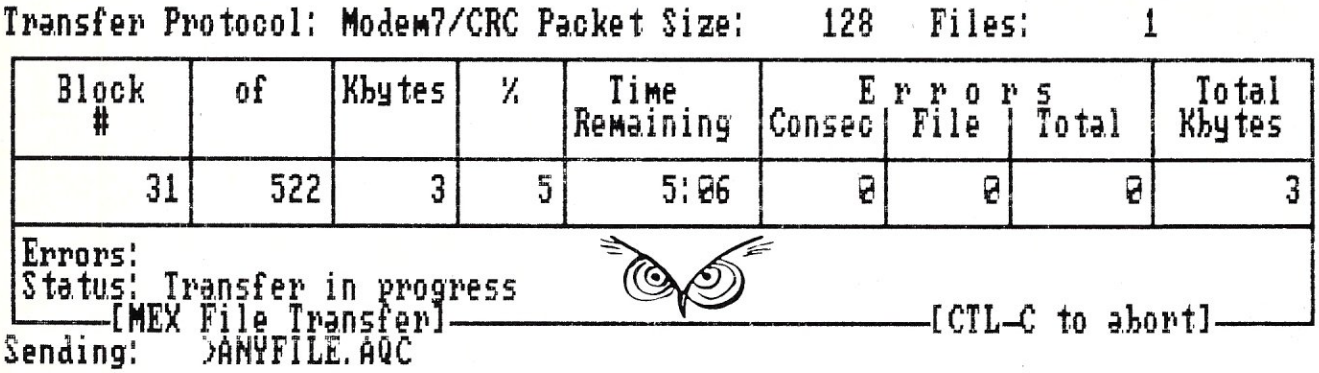
It does support something resembling a medium-memory model (multiple segments of code but the data and stack must fit in one 64K segment). The medium model, or what VenturCom calls code mapping, is achieved completely by the linker, not the compiler. Rather than generating code for *far* calls, when the code size exceeds 64K, the linker arranges for a call to be vectored into low memory, whence it issues a software interrupt to be vectored into another segment. This has several negative consequences. The most serious is that the debugger, adb, flat out doesn't work well with code-mapped programs. You cannot set a breakpoint at an address outside the current code segment, and breakpoints in the current code segment can cause spurious breakpoints to be issued. And, of course, the overhead with this scheme is nontrivial.

SCO V comes with libraries for cross-compiling DOS programs under XENIX; there is no such beast from VenturCom, and VENIX comes out a significant loser here. VENIX V has a new C compiler that supports enum and void and some of the other post-K & R C enhancements.

INSTALLATION

Installing VENIX V went smoothly. Compared to the daily things you have to know how to do to run UNIX on your own machine, the installation is a snap, being completely menu-driven. VENIX lets you tell it the head and cylinder layout before you start in case you have a nonstandard kind of hard disk. It did not ask for the write precompensation cylinder, however. If you are running it on a PC rather than an AT with a hard disk that uses write precompensation, you might have a serious problem installing VENIX. In contrast to SCO, which publishes a list of supported controllers, neither VenturCom nor Unisource supply a list of controllers on which VENIX will work. If in doubt you'd

What you see is what you get . . . and send!


Transfer Protocol: Modem7/CRC Packet Size: 128 Files: 1
Block # | of | Kbytes | % | Time Remaining | Errors: Consec | File | Total | Total Kbytes
31 | 522 | 3 | 5 | 5:06 | 0 | 0 | 0 | 3
Errors:
Status: Transfer in progress
[MEX File Transfer]
[CTL-C to abort]
Sending: >ANYFILE.AQC

Announcing Version 1.6 of **MEX**, the communications software with a view from the top. Regardless of your level of sophistication, **MEX** can put you on top of the data transfer game and keep you there. For the executive on the go, our new pull-down transfer screen and easy-to-use menus reduce the complexities of modem communications to a few keystrokes. For the advanced user, **MEX**'s greatly enhanced script processor offers a complete programming language for development of highly secure custom applications. If communication is money in your business, **MEX** may be the best investment you make this year.

Two options avaiiable:

MEX-PC is the most complete modem software you can buy. Allows you to switch between menu-driven and command-driven communications at will. Makes full use of Hayes AT command set, with overlays available for most other modems. Features include: complete script processor programming language; user-definable keystrings; auto-dial and auto-baud-set phone libraries; all popular protocols, including MODEM-7 batch transfers **$59.95***

XMODEM CRC/XMODEM CHECKSUM/KERMIT/COMPUSERVE A/128 or 1K BLOCKS

MEX-PAC — All the features of **MEX-PC**, plus:

()A remote module that allows you to run your office computer from home, and vice versa; and

()Terminal emulation that lets your PC masquerade as a DEC VT52/100 or Televideo 925 terminal for on-line communication with mainframes. **$99.95***

Versions available for IBM-PC and compatibles, Tandy 2000 and most CP/M machines.

MasterCard, VISA welcome

Give us a toll-free call at 1-800-NITEOWL

In Wisconsin, 1-414-563-4013

* plus shipping; Wisconsin residents add 5% sales tax.

NightOwl Software, Rt.1 Box 7, Ft. Atkinson, WI 53538

SYSTEMS or BOARDS, ...YOU CHOOSE

INTRODUCING THE LO-MAX, LDP'S NEWEST MULTI-USER SYSTEM.

LO-MAX is **LO**w on price and **MAX**imum on performance. LO-MAX is LDP'S newest system based on the THUNDER/PLUS. LO-MAX has the following advanced features:

- 10 Mhz 80186 Microprocessor.
- 512K bytes of no-waitstate dynamic RAM.
- Supports two users.
- CC-DOS version 4.1 runs both MS-DOS and CP/M-86 programs.
- 7 slot chasis for future expansion.
- 20 Mbyte Harddisk drive.
- SCSI interface for expansion.

OPTION:

- 1 Mbyte of no wait-state dynamic RAM.
- 8087 option.
- 1.2 Mbyte PC/AT compatible 5¼" floppy drive.
- 40 to 140 Mbyte harddisk options.
- Five user support.

S100 BOARDS/SYSTEMS

CPU's:
80186 SBC
80286 8/10MHZ

MEMORY:
STATIC RAM 1MBYTE
DYNAMIC RAM 2MBYTE

INTERFACE:
DUAL HARDDISK/FLOPPYDISK CONTROLLER
8 PORT SERIAL BOARD
SCSI ADAPTER WITH SERIAL PORTS
MULTIFUNCTION SYSTEM SUPPORT BOARD
PC COMPATIBLE GRAPHICS FOR S100 BUS SYSTEMS

SYSTEMS:
2 TO 18 USER SYSTEM

All these features make this system too great a value to pass up. If you need a multiuser system with the performance necessary to support multiple users buy our S100 BUS system, designed for the start to run multiple users.

ALL THESE FEATURES FOR AN UNBELIEVABLY LOW PRICE OF *ONLY $3695.00*

NEW S100 BUS SCSI ADAPTER

Open your system to the world of SCSI peripherals. The LDP-SCSI adapter board enables you to add numerous peripherals that use the industry standard SCSI interface standard to your S100 bus system. Today's high density harddisk drives are offering SCSI interfaces to improve performance and increase the data transfer rates, many tape drives units now support the SCSI standard. In addition to the SCSI interface, the board includes a battery protected CLOCK CALENDAR, and four RS232 ports capable of communicating at baud rates up to 38400 BAUD. Move your system into the future with the LDP-SCSI. ***ONLY $375.00***

THUNDER/PLUS

THUNDER/PLUS is a complete 16 bit S100 computer on one card. It contains all of the following as standard features:

- 512K bytes of no wait-state dynamic RAM expandable to 1 MEGABYTE on board.
- Two RS232 serial ports capable of communicating at up to 19200 BAUD.
- Centronics compatible parallel interface.
- Floppy disk controller capable of controlling a mix of 8", 5¼" and 3½" floppy drives including the PC/AT 1.2 Mbyte high density 5¼ drives.
- 10 MHZ 80186 Microprocessor.
- CONCURRENT-DOS version 4.1.

ALL THE PERFORMANCE AND FEATURES YOU REQUIRE IN A 16 BIT MICRO AT AN *AFFORDABLE $1195.00*

THUNDER/PLUS and LDP-SCSI make an unbeatable 5 user computer system

NEW LOWER STATIC RAM PRICES !!!

MEGA-S-RAM static RAM board available with up to 1 Megabyte of fast static RAM.
16K bytes ***$349*** 256K bytes ***$495*** 1/2 Mbyte ***$995*** 1 Mbyte ***$1749***

LDP

LOMAS DATA PRODUCTS, INC.

Call or write for a complete catalog of our S100 BUS BOARDS AND SYSTEMS.

(617)460-0333
182 Cedar Hill Street, Marlboro, MA 01752

better call Unisource.

A major improvement over previous releases is the ability to map out bad tracks under VENIX's control. I entered no bad tracks explicitly because I had marked them with a low-level format of the hard disk. VENIX found them and remapped them. It would have been nice, though, for it to have printed a screen message: "Bad track found at cylinder ___ head ___." Instead, when it found a bad track, it printed nothing on the screen. It was only by running a program after VENIX was installed that I verified it had found all the right bad tracks.

DOCUMENTATION

VENIX V documentation is a real surprise. It's nicely typeset with a wonderfully readable layout. Of course, it's still that same old UNIX documentation you get with any version of UNIX. The manual set has the usual organization into commands, system calls, subroutines, file formats, and so on—unlike SCO, which saw fit to reorganize its manual set for SCO V into sections that will bewilder both experienced UNIX users and novices—quite a feat! VENIX V uses the now mandatory IBM PC-size three-ring binders, without slipcases (which I despise anyway). There is one absolute treasure trove in the VENIX documentation. The UNIX file system is somewhat complicated, and when it becomes inconsistent, all hell breaks loose. The file system is checked by a program called fsck, and when it reports trouble, you may have a tiny problem or a major mess on your hands. VenturCom gets 64K gold stars for documenting all the fsck error messages. This could save your file system some day, and very few UNIX manual sets have it.

SUPPORT

Support for VENIX is a mixed bag. At one time VenturCom simply would not talk to end-users, and you could get support only from Unisource. As VenturCom is now selling VENIX directly, this has changed.

Unisource has a toll-free number, and the support people are helpful souls—knowledgeable about UNIX. But, alas, they have one hand tied behind their backs. It seems Unisource does not have a full source license. Problems involving bugs or anything requiring consulting the source code must go to VenturCom. This arrangement is cumbersome. The one important question I had about VENIX V, relating to what VenturCom's strange concept of autosync meant, was answered effectively by the Unisource support person consulting VenturCom.

Not all problems are solved so simply. Under VENIX 2.0 the disk driver I was supplied with was buggy. I was decidedly unhappy because I had followed Unisource's recommendations on what hardware to buy to ensure compatibility. The bug in the disk driver prevented making a file system after the bad blocks had been marked by a low-level format. Unisource finally decided that letting me fix it myself was more fun than listening to me scream at them. In exchange for source to the driver, the company got my fix. In this instance support worked the other way around: I gave the company my support! Note that buying direct from VenturCom may not be a practical solution to the support problem because it's only Unisource that supports many of the clones and different disk controllers.

Unisource sells a variety of third-party software for VENIX that covers the usual range of applications—for example, word processing, database management, and spreadsheets. Unfortunately, the support is not very deep. Typically a given application is available from only one vendor. A quick scan of any trade magazine targeted at UNIX reveals almost no trademark notices for VenturCom but an abundance of notices that XENIX is a trademark of Microsoft. If you would like to run UNIX on a PC-compatible just to learn UNIX, then this won't matter. (But if you have an AT and are in this situation, you'll surely buy the Microport System V because the cost is less than half that of VENIX!) If you're looking for a UNIX that will run a variety of application software, though, the support for XENIX is so much better that it's virtually no contest. As a user I don't want Unisource supporting me with application software—I want the marketplace to support me. This is not happening. Time and again I've contacted vendors asking if they will have a port of this or that product to VENIX. The answer is usually no. It makes me wonder how VenturCom stays in business.

At one time there was talk that VENIX V would use a format for executables that would be compatible with XENIX. That would have been an excellent move. By not making it, VenturCom missed a very large boat. Not only are VENIX V binaries incompatible with XENIX, or any other form of UNIX, performance of binaries from VENIX 1.18 and 2.0 under VENIX V is erratic. Some work, some do not, and the manual states that there is no guarantee they will work. This is frustrating. SCO went to enormous trouble to preserve backward compatiblity. VARs and those trying to sell software as binaries on standard media find the backward compatibility of XENIX extremely helpful. If VenturCom were going to ditch backward compatibility of its binaries, it should have bitten the bullet and adopted a XENIX-compatible format for binaries.

CONCLUSION

My final verdict is an enthusiastic thumbs sideways. I trust VenturCom to produce the finest UNIX kernel money can buy on PC-compatible equipment, but the poor position of VENIX in the marketplace as far as third-party product support, the lack of a large-memory model C compiler, and SCO's DOS cross-libraries make it SCO's ballgame to lose. VENIX works extremely well. If you don't need third-party software, and especially if you have some reason to mistrust XENIX, VENIX is an excellent piece of work and as "nothing but UNIX" is outstanding. If there are skeletons in the SCO closet, of course, all bets are off. §

Jim Rosenberg is president of Amanuensis Inc., a small software and consulting company near Pittsburgh. He's been programming computers since 1965 and is the author of Qbax, an incremental backup program for CP/M. He can be reached by Email at: uucp: {decvax!idis,allegra,seismo!cmcl2!cadre}!pitt!amanue!jr; CIS: 71515,124; WELL: jer.

PRODUCT INFORMATION

VENIX System V
A port of UNIX System V, Release 2, for IBM XT/AT machines and compatibles.

Unisource Software
23 East St.
Cambridge, MA 02141
(800) 343-5504
$1,150 (two-user license)
$1,350 (eight-user license)

VenturCom
215 First St.
Cambridge, MA 02142
(617) 661-1230
$1,195 (two-user license)
$1,395 (eight-user license)

UNIX is a trademark of AT&T Bell Labs
VENIX is a trademark of VenturCom
XENIX is a trademark of Microsoft

A POWERFUL UTILITY

For under $20, get a powerful programmer's utility that will bring you a year of:

- **hot technical tips** from experts like Tom Swan, Ray Duncan, Michael Abrash, William Hunt and Rex Jaeschke.
- **guidance for consultants** from Paul Barkley— **industry news** from Frank Greco and Hal (*DTACK Grounded*) Hardenbergh.
- **valuable and efficient code** in ASM, C, Pascal, BASIC
- **expert tips** on operating systems, the 386, graphics drivers, the EGA, and more!

RESPECTED BY PROFESSIONALS

"For hot programming tips look to... the excellent but relatively little known **Programmer's Journal**."
Peter Norton —*Inside the IBM PC*

"...stuffed with useful information, it definitely deserves a look."
Ray Duncan—*DDJ* 6-86

"I really love this magazine!"
Harry Miller —Editor, *PC WORLD*

GET ONE FREE!

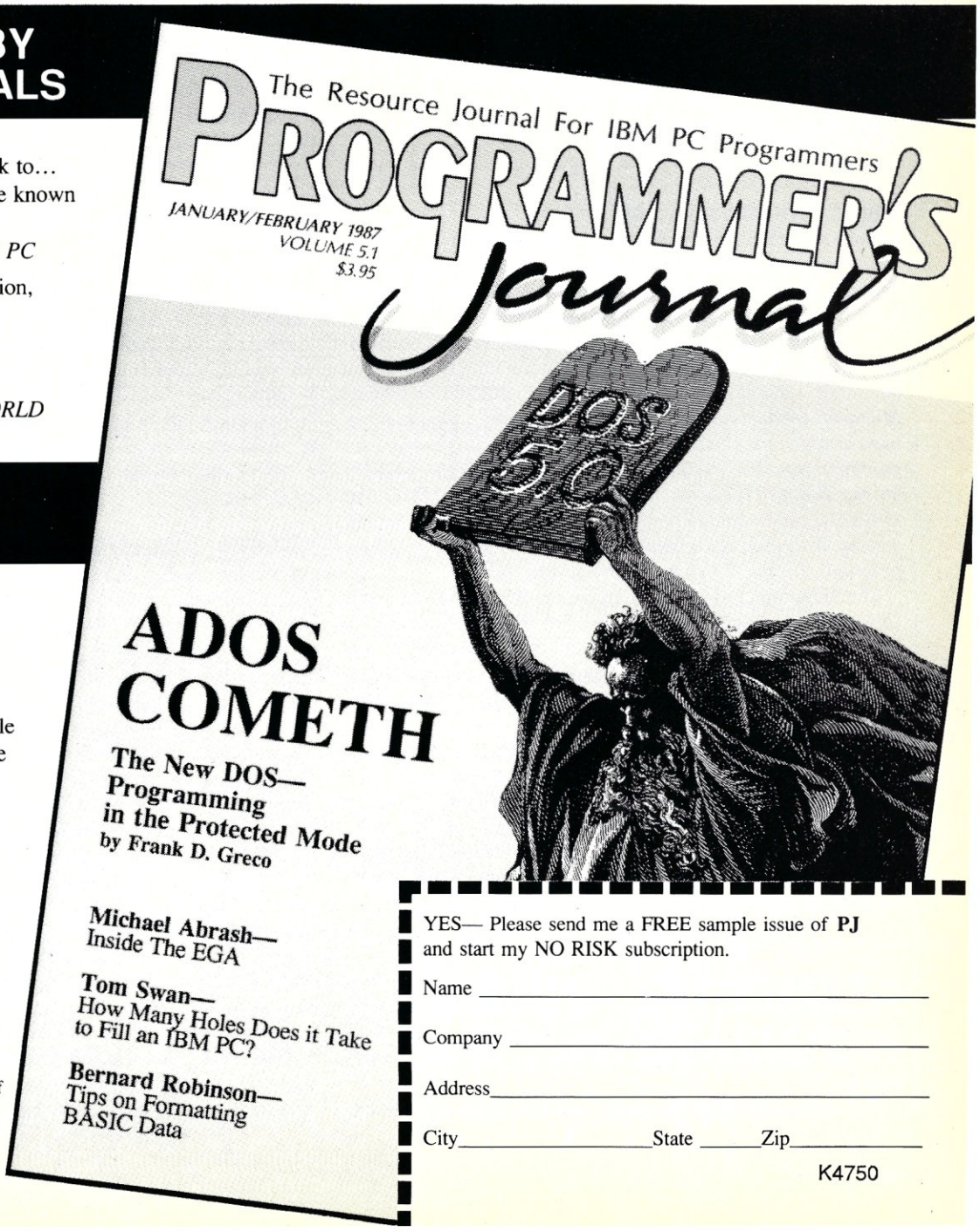
To see what ***PJ*** has to offer just do <u>one</u> of the following:

- call us at (503) 484-2162
- return the coupon at right

We'll send you a **FREE** sample copy of our latest issue, reserve a 1 year subscription (6 issues in all) and invoice you for $19.95. If **PJ** is not the utility you need, simply write "CANCEL" on the bill and return it. Keep your free issue and owe nothing.

Programmer's Journal
29 W. 29th Suite 5
Eugene, OR 97405

Please allow 4-6 weeks for delivery of first issue. Offer good in U.S. only. Foreign subscriptions must be prepaid in U.S. funds. Can/Mex $29.95, elsewhere $39.95.


The Resource Journal For IBM PC Programmers
PROGRAMMER'S Journal
JANUARY/FEBRUARY 1987
VOLUME 5.1
$3.95
DOS 5.0
ADOS COMETH
The New DOS— Programming in the Protected Mode
by Frank D. Greco
Michael Abrash— Inside The EGA
Tom Swan— How Many Holes Does it Take to Fill an IBM PC?
Bernard Robinson— Tips on Formatting BASIC Data

YES— Please send me a FREE sample issue of **PJ** and start my NO RISK subscription.

Name _____

Company _____

Address _____

City _____ State _____ Zip _____

K4750

Program Interfacing to Microsoft Windows

by William Wong

Part III— Windows Within Windows

WRITING WINDOWS APPLICATIONS

Windows applications cannot be written in MASM or C alone. This is because the linker must understand Windows objects (for example, icons, dialog boxes) and linking to Windows is through a special call linkage, not an INT *instruction.*

To write a windows application program, you need the Windows Toolkit ($500 list, but discounted to as little as $309) and the latest version of either Microsoft C, MASM, or Pascal. Microsoft conducts seminars for windows application programmers at which the Windows Toolkit package is distributed. Programmers who but the Toolkit without attending the seminars are handicapped because the manuals contain little theory. This series of articles should assist programmers who buy the Toolkit without attending the seminars to write Windows application programs. It should also help seminar attendees.

The Whitewater Group (Technology Innovation Center, 906 University Pl., Evanston, IL 60201; (312) 491-2370) is expected shortly to introduce Actor, a lower-cost alternative to the LMicrosoft Windows Toolkit. However, its documentation is poorer than that of the Toolkit because it deals primarily with the Actor programming language.

This series of articles has a twofold purpose. First, its intent is to give readers considering purchasing the Microsoft Windows Toolkit and writing a windows application an insight into what is involved. Its second purpose is to help readers who have purchased the Toolkit or attended a Microsoft Windows seminar to write a typical application. Some sample programs will be provided after sufficient groundwork has been laid.

As its name implies, Microsoft Windows is a windowing system. This article addresses windows within Windows. Although windows are only part of Windows, they are a central part—every Windows application normally has at least one window associated with it.

The support functions, messages, and overall structure of the Window system are relatively complex. This article is divided into two main sections: Coroutines and Classes, and Window Functions. The first section describes the relationship between Windows, applications, and windows. The second section describes some of the Windows functions that you use to create and manipulate windows. Part IV covers some of the messages passed among applications related to the creation and manipulation of windows.

There are two ways in which you can try out these functions and messages. The first is to use the Microsoft Windows Toolkit, which supports C, Pascal, and assembly language. The second is to use Actor from Whitewater Group ([312] 491-2370). Actor is an object-oriented language and development system based on Smalltalk, which was developed at XEROX. Smalltalk is the window-based system from which much of Windows was developed. Actor is actually a great way to experiment with Windows because the edit-compile-link cycle of the Toolkit languages can be time consuming.

COROUTINES AND CLASSES

As mentioned in the previous article, Windows uses a cooperative coroutine form of multitasking. Just in case there is any confusion, each instance of a Windows application is a coroutine with an associated program stack. A second instance of a Windows application is a second coroutine. The two coroutines share several things, including code and window classes. Each coroutine has its own stack and data segment, however.

Communication between Windows and applications is performed using functions supplied by Windows. The communication and coordination is done using the Window message-passing system. However, the message-passing system is linked via windows and coordinated via the application task.

Each application task starts out by initializing itself and then entering a tight loop to handle message dispatch within the application. All the work is done by the rest of the application, which has been set-up to receive the messages dispatched by the main portion of the application. The messages are kept in an application queue that can be fed from several sources, including the application, other applications, and Windows.

The dispatch function lets Windows send the processed message to the proper window functions. Window functions for a particular application use the same stack as the dispatch func-

tion. Windows and its messaging system become more important as the number of windows an application controls increases. Windows keep track of which messages are destined for a particular window and also how other Windows resources, such as the mouse, are handled with respect to a window.

Each window in Windows is an instance of a window class. A window class holds common attributes about a window and the reference to the window function that handles each message sent to a window. Figure 1 shows how a window class relates to an instance of a window. In this example, the two application tasks are instances of the same application, and a single window class supports three windows.

A new class is normally used for each window that has a different function—for example, a terminal program might have a window class for the terminal window, one for the parameter setup window, and another for transferring files.

Often, there is one window instance per class, but the class system provides an efficient way to support both multiple window instances within an application and multiple copies of an application.

The window function within a window class can send messages to itself or to any other window in the system. This can be done directly using a send-message type Windows function or indirectly by calling a Windows function that sends a message. For example, a tiled window that reverts to icon form would cause other tiled windows to receive a message to fill up the space vacated by the iconized window.

Instances of windows are grouped into three distinct types: tiled, pop-up, and child windows. Tiled windows are those that fill up the work area, don't overlap, and can be converted to icons. Pop-up windows are ones that overlap other windows. A child window is one that is contained within another window, which can be a tiled, pop-up, or another child window (which would be contained within another window). Figure 2 shows some of the possible relationships between windows. Pop-up and child windows cannot be converted to icons. Any window can be hidden, and any window, viewable or hidden, can receive messages.

By the way, all the messages discussed here are relatively simple and consist of three parts: a message number, a 16-bit parameter, and a 32-bit parameter (parameters are optional). A message function in the message class is called with these items plus the handle for the instance of the window receiving the message.

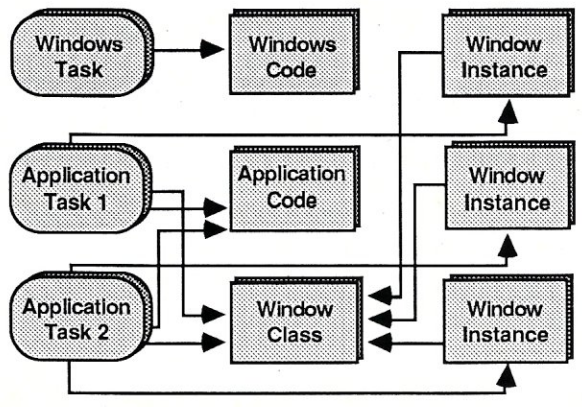
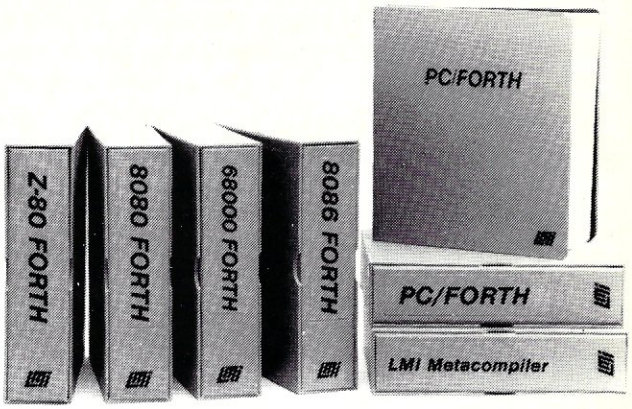


Figure 1. Windows, classes and instances.

TOTAL CONTROL with *LMI FORTH*™



For Programming Professionals:

an expanding family of compatible, high-performance, Forth-83 Standard compilers for microcomputers

For Development: Interactive Forth-83 Interpreter/Compilers

- 16-bit and 32-bit implementations
- Full screen editor and assembler
- Uses standard operating system files
- 400 page manual written in plain English
- Options include software floating point, arithmetic coprocessor support, symbolic debugger, native code compilers, and graphics support

For Applications: Forth-83 Metacompiler

- Unique table-driven multi-pass Forth compiler
- Compiles compact ROMable or disk-based applications
- Excellent error handling
- Produces headerless code, compiles from intermediate states, and performs conditional compilation
- Cross-compiles to 8080, Z-80, 8086, 68000, 6502, 8051, 8096, 1802, and 6303
- No license fee or royalty for compiled applications

For Speed: CForth Application Compiler

- Translates "high-level" Forth into in-line, optimized machine code
- Can generate ROMable code

Support Services for registered users:

- Technical Assistance Hotline
- Periodic newsletters and low-cost updates
- Bulletin Board System

Call or write for detailed product information and prices. Consulting and Educational Services available by special arrangement.

Laboratory Microsystems Incorporated
Post Office Box 10430, Marina del Rey, CA 90295
Phone credit card orders to: ***(213) 306-7412***

Overseas Distributors.
Germany: Forth-Systeme Angelika Flesch, Titisee-Neustadt, 7651-1665
UK: System Science Ltd., London, 01-248 0962
France: Micro-Sigma S.A.R.L., Paris, (1) 42.65.95.16
Japan: Southern Pacific Ltd., Yokohama, 045-314-9514
Australia: Wave-onic Associates, Wilson, W.A., (09) 451-2946

One reason for all these messages—in addition to general interaction between windows, the mouse, and the keyboard—is because of the way in which Windows requires windows to repaint themselves when things such as removing a pop-up window causes part of a window to become visible again or when a tiled window changes size or position. An application must then redraw the affected areas based upon the parameters specified in the messages.

Although this seems simple, the number of different message numbers is large and the parameters are specific to the message. The method of maintaining a window is also critical to an application because the speed at which a window can be repainted can affect the overall performance of the system. The amount of space required is also a consideration—for example, a text screen for a terminal program or simple text editor would require the application to keep the text to be drawn in the window in addition to keeping it on the screen. The redraw overhead and memory costs are minimal. A CAD program might maintain its database as a list of objects that may take a minimal amount of space, but it could take a while to redraw a portion of a window if the entire database must be scanned. One alternative here is to keep a copy of the window in bit-map form, so redrawing a portion of the screen is quicker. This is a time/space trade-off that is more critical here than in the previous example.

The following sections address the functions and messages for window manipulation.

WINDOW FUNCTIONS

The functions described in this section are available from Windows and are linked by name using the dynamic linking facility provided with Windows. The functions are shown using C syntax.

The first set of functions are:

```
WORD   WinMain        ( hInstance, hPrevInstance,
                        lpCmdLine, nCmdShow )
LONG   WndProc        ( hWnd, wMsg, wParam,
                        lParam )
LONG   DefWindowProc  ( hWnd, wMsg, wParam,
                        lParam )
```

WinMain is defined, not by Windows, but within the application. This function is called by Windows to start an application, and it should exit only when the application is ready to terminate. The resulting code is the program termination code. *HInstance* is the task handle for the particular instance of the application. The *hPrevInstance* handle is nonnull if the application has already been started once before. This is included because a second instance can use some of the resources already loaded by the initial application instance. *LpCmdLine* is a long pointer to the string used to start the application. This is one way an application can start a copy of itself and pass it a different set of parameters. The second task operates independently. This is significantly different from multiple windows. The *nCmdShow* parameter should be passed to *ShowWindow,* described later in this section.

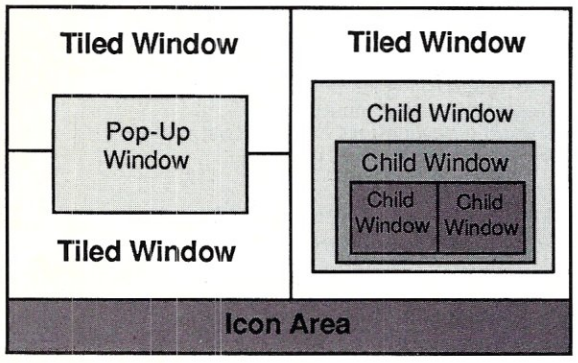


Figure 2. Relationships between windows.

The *WndProc* function is actually a prototype. The name of the function can change, as shown in the previous article. In fact, there will probably be several different instances within an application, depending upon the number of different window classes there are. Each window class has a *WndProc* associated with it.

DefWndProc is a Windows-supplied function that should be called by a *WndProc* function when it encounters a message that it can handle itself. The *hWnd* parameter is the handle of the window receiving the message, the message number is *wMsg,* and the parameters associated with the message are *wParam* and *lParam.*

The following set of functions is used within the *WinMain* function dispatch loop:

```
BOOL   GetMessage         ( lpMsg, hWnd,
                            wMsgFilterMn,
                            wMsgFilterMax )
BOOL   PeekMessage        ( lpMsg, hWnd,
                            wMsgFilterMin,
                            wMsgFilterMax,
                            bRemoveMsg )
BOOL   TranslateMessage   ( lpMsg )
BOOL   TranslateAccelerator ( hWnd, hAccTable,
                              lpMessage )
LONG   DispatchMessage    ( lpMsg )
VOID   WaitMessage        ( )
```

The dispatch loop shown in Figure 3 consists of getting a message using *GetMessage*, or of taking a quick peek using *PeekMessage*, then translating the message using either *TranslateMessage* or *TranslateAccelerator*, followed by a *DispatchMessage* call. The message structure is referenced by *lpMsg*, which is a long pointer. The *hWnd* handle can refer to a specific window associated with an application; otherwise, it can be null, which tells Windows to get a message for any window associated with the application task instance. The *wMsgFilterMin* and *wMsgFilterMax* parameters are used to allow an application to check for specific types of messages, such as mouse movements or keyboard input. An application need not process a message on a first-come-first-served basis.

The *bRemoveMsg* flag can be set to true, in which case it is the same as a *GetMessage*. A false value leaves any message found within the application queue, and it can be obtained again using another *GetMessage* or *PeekMessage* call. The accelerator table is referred to using the *hAccTable* handle, which should be obtained when the table is loaded. This Windows function will be described in a subsequent article.

The *TranslateMessage* and *TranslateAccelerator* functions are used to convert the message and associated parameters to something that is more usable by most applications. The following article on window messages will clarify this. An example is a

conversion of a key-down and a key-up message (two different messages) into a key-code message and a dead-key message because most applications need only to see when a key is pressed. Figure 4 shows how *TranslateAccelerator* extends the dispatch loop.

Finally, the *DispatchMessage* function lets the *WndFunc* for the corresponding window instance actually process the message. *DispatchMessage* returns when the message has been processed, and the result is returned by the selected *WndFunc*.

WaitMessage is used when there is no other work to be done by an application and it is ready to wait for another message. This is not necessary if *GetMessage* is used because it waits all by itself. *WaitMessage* is normally used when *PeekMessage* is used to examine incoming messages.

Note that the message-passing functions must be used, but the translate functions are optional. Also, an application can substitute or augment the Windows-supplied functions.

Messages have a cursor position and time stamp associated with them. These can be obtained for the last message read using *GetMessage*. The functions are:

```
POINT   GetMessagePos  ( )
TIME    GetMessageTime ( )
TIME    GetCurrentTime ( )
```

The *GetCurrentTime* function uses the same units as the *GetMessageTime* function does and can be called from anywhere within the application. *GetCurrentTime* is not affected by the *GetMessage* function. The current time is useful in determining how long it has been since the message was read.

Messages are sent by Windows as necessary, but applications can send messages to their own windows, other tasks, or other task's windows. The following functions are used for this purpose:

```
BOOL PostMessage      ( hWnd, wMsg, wParam,
                        lParam )
BOOL PostAppMessage   ( hTask, wMsg, wParam,
                        lParam )
```

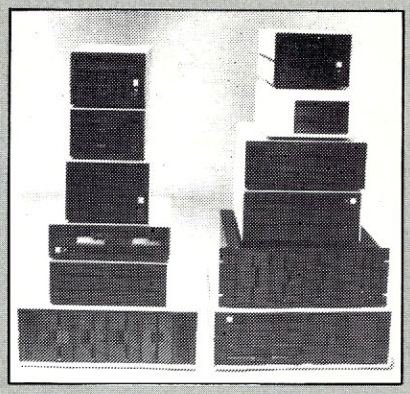
```
while (GetMessage((lpMsg)&Msg,NULL,0,0))
{
      Translate Message((lpMsg)&Msg);
      Dispatch Message((lpMsg)&Msg);
}
exit (Msg.wParam);
```

Figure 3. WinMain dispatch loop.

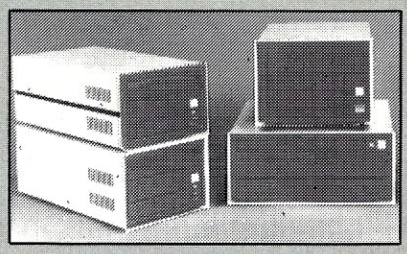
```
while (GetMessage((lpMsg)&Msg,NULL,0,0))
{
    if (Translate Accelerator (hWindow,
       hAccel, (lpMsg)&Msg)==0)
    {
        Translate Message((lpMsg)&Msg);
        Dispatch Message((lpMsg)&Msg);
    }
}
exit (Msg.wParam);
```

Figure 4. WinMain dispatch loop with accelerator support.

ENCLOSURES & POWER SUPPLIES



for FLOPPY WINCHESTER TAPE DRIVES
from $100



for SINGLE BOARD COMPUTERS
from $125

for IBM & S-100 SYSTEMS from $200

*1 piece; prices lower in quantity.
(Disk drives not included)

Call or write for free catalog

INTEGRAND RESEARCH CORP.

8620 Roosevelt Ave. • Visalia, CA 93291
209/651-1203
TELEX 5106012830 (INTEGRAND UD)

IBM IM International Business Machines

```
VOID PostQuitMessage ( nExitCode )
```

The *PostMessage* function is used to send or add a message to a window. The message is placed into a tasks application message queue and must be removed using *GetMessage* or *PeekMessage* within the respective task. The message is sent to all tiled and pop-up windows in the system if the value of *hWnd* is FFFF hex. This is a useful broadcast feature.

The *PostAppMessage* function puts a mesage in the application queue, but the window handle is set to null. The function requires knowledge of the task handle, *hTask*. A task's handle is the *hInstance* parameter in the initial *WinMain* function. The Boolean result of these functions is false if the message is not posted in an application queue.

The *PostQuitMessage* function is normally called when an application is terminating and performs the same function as:

```
PostAppMessage ( myTask, WM_QUIT,
                 nExitCode, 0L )
```

where *myTask* is the task handle for the task invoking the *PostQuitMessage* function.

Messages can be sent directly to a window, even within another task, if the window handle is known. The following functions perform this task:

```
LONG  SendMessage  ( hWnd, wMsg, wParam,
                     lParam )
VOID  ReplyMessage ( lReply )
```

The *SendMessage* function causes the *WndProc* function of the corresponding window specified by *hWnd* to be called with a message having the associated parameters. *WndProc* is invoked as a function call if the window is owned by the calling task; otherwise, a coroutine task switch is performed. The result of *SendMessage* is the value returned by the invoked *WndProc* function. The *WndProc* function can also return a value to *SendMessage* using the *ReplyMessage* function. The returned value of *WndProc* is ignored in this case. *ReplyMessage* is handy if the message request can be indicated immediately but additional processing is necessary. Only the initial *ReplyMessage* is recognized.

Message numbers can be predefined within an application, and cooperating applications can have predefined messages. A better method of having cooperative processes exchange messages, however, is to using the following function:

```
WORD  RegisterWindowMessage ( lpString )
```

This function returns a unique number in the range C000 to FFFF hex if the reference string has not been previously registered. The same message number is returned if the string matches a registered string. The message registry only grows and is initially empty when Windows starts. The message number returned by this function will normally be different between invocations of the Windows system because the number is dependent upon the order of message registration.

WINDOW CLASS FUNCTIONS

The next set of functions is related to window classes and creation of instances of windows.

```
BOOL RegisterClass  ( lpWndClass )
WORD GetClassName   ( hWnd, lpClassName,
                      nMaxCount )
WORD GetClassWord   ( hWnd, nIndex )
LONG GetClassLong   ( hWnd, nIndex )
WORD SetClassWord   ( hWnd, nIndex, nValue )
LONG SetClassLong   ( hWnd, nIndex, lValue )
LONG CallWindowProc ( lpPrevWndFunc, hWnd,
                      wMsg, wParam, lParam )
```

A class is registered by calling *RegisterClass* with a long pointer to an initialized data structure of type *WndClass*, which was described in the first article. The major items in the structure are the class name and the *WndFunc* reference. A class is identified by the class name. The result of *RegisterClass* is true if the class is registered, and a class may superscede a previous class with the same name. The *RegisterClass* function fails if there is insufficient space. *RegisterClass* copies the contents of the *WndClass* structure, so the subsequent set of functions is necessary to change or examine Window's copy of the data structure. The *nIndex* values are:

```
GCW_HBRBACKGROUND
GCW_HCURSOR
GCW_HICON
GCW_HINSTANCE
GCW_CBWNDEXTRA
GCW_CBCLSEXTRA
GCW_STYLE

GCL_MENUNAME
GCL_WNDPROC
```

The *CallWindowProc* function is used to create window subclasses. It is first necessary to use *GetClassLong* with *GCL_WNDPROC* to get the current *WndProc* for a window class. *SetClassLong* is then used to update the *WndProc* function. This new function would have access to the retrieved *WndProc* reference and would now field all messages for the new window instances. The new *WndProc* can pass all messages it does not process itself to the old *WndProc* function using the *CallWindowProc* function, where the first parameter is the old *WndProc* address. The method of message processing is arbitrary, and many types of subclassing and inheritance can be implemented. Unlike instances of windows, classes do not go away.

WINDOW INSTANCE CREATION FUNCTIONS

Window creation can proceed only after a window class has been registered. The following functions are used to create, manipulate, and destroy instances of a window class.

```
HANDLE CreateWindow   ( lpClassName,
                        lpWindowName, dwStyle,
                        nX, nY, nWidth,
                        nHeight, hWndParent,
                        hInstance, lpParam )
BOOL   DestroyWindow  ( hWnd )
WORD   GetWindowWord  ( hWnd, nIndex )
LONG   GetWindowLong  ( hWnd, nIndex )
WORD   SetWindowWord  ( hWnd, nIndex, nValue )
LONG   SetWindowLong  ( hWnd, nIndex, lValue )
```

CreateWindow references a window's class by name. The window name is used for windows with caption bars. *DwStyle* can be a combination of the following:

```
WS_TILED
WS_POPUP
WS_CHILD
WS_ICONIC
```

```
WS_BORDER
WS_CAPTION
WS_DLGFRAME
WS_SYSMENU
WS_SIZEBOX
WS_VSCROLL
WS_HSCROLL
WS_VISIBLE
WS_DISABLED
WS_CLIPCHILDREN
WS_CLIPSIBLINGS
WS_TILEDWINDOW
WS_POPUPWINDOW
WS_CHILDWINDOW
```

WS_TILED, WS_POPUP, and *WS_CHILD* specify the type of window. *WS_ICONIC* can be used only with *WS_TILED* and indicates that the window should initially be iconic. *WS_BORDER, WS_CAPTION, WS_DLGFRAME, WS_SYSMENU, WS_SIZEBOX, WS_VSCROLL*, and *WS_HSCROLL* enable the respective attributes around the window. *WS_VISIBLE* and *WS_DISABLED* set a window's initial respective attribute and can be applied to any type of window. *WS_CLIPCHILDREN* and *WS_CLIPSIBLINGS* specify how a window is to be clipped with respect to other windows when drawing is performed. *WS_TILEDWINDOW* and *WS_POPUPWINDOW* are predefined combinations for *WS_TILED, WS_CAPTION, WS_SYSMENU,* and *WS_SIZEBOX* and *WS_POPUP, WS_BORDER,* and *WS_SYSMENU,* respectively. *WS_CHILDWINDOW* is an alias for *WS_CHILD.*

The *nX* and *nY* parameters are a window's position with respect to the parent window, which is the screen in the case of a tiled window. The *nWidth* and *nHeight* parameters specify a window's size. Tiled windows are special in that all these parameters are ignored—except *nY,* which should be a *ShowWindow* function parameter if *dwStyle* contains *WS_VISIBLE.*

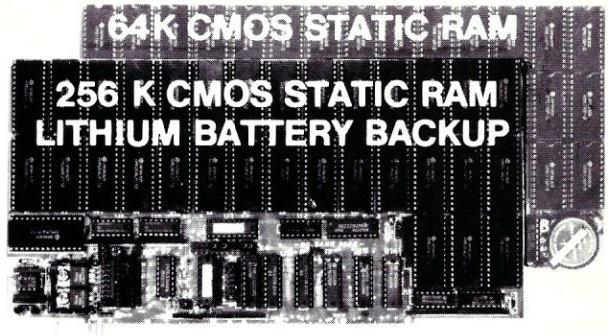
The parent of a window, specified by *nWndParent*, must be null for tiled windows, can be null for pop-up windows, and cannot be null for child windows (that is, child windows must have a parent). The menu handle, *hMenu*, is that specified by the window class if the parameter is null. A different menu can be used if one is specified here. Of course, *WndProc* for the class would still have to understand the messages generated by such a menu if it is enabled. This parameter is used by child windows to specify a special identification number that should be unique for each child window with respect to a parent.

The *hInstance* parameter is used to indicate the owner of the window. The parameter is the same task handle as is passed to the *WinMain* function. Having a task create a window for another instance is possible. The *lpParam* is an arbitrary number that can be used by the *WndProc* function to identify attributes for a particular window. The parameter is large enough to contain a handle or pointer to a data structure if more information is necessary.

The *CreateWindow* function returns null if a window cannot be created—for example, if the window class has not been previously registered. Otherwise, the result is the window handle, which can be used to send messages to the window. The window handle can also be used to return a window and its resources to Windows using the *DestroyWindow* function.

GetWindowWord, GetWindowLong, SetWindowWord, and *SetWindowLong* are used to examine or modify Windows' internal window structure. The *nIndex* parameter can be one of the following:

HIGH PERFORMANCE RAM



✓ COMPARE

	8/16 BIT DATA	Bank Selection MPM, OASIS, CROMIX-D	10 Megahertz Speed	Software Write Protect	Battery Backup Option
Compupro Ram 22	✓	NO	✓	NO	NO
Octagon 256K	✓	NO	✓	NO	NO
Cromemco 256KZ II	✓	NO	NO	NO	NO
Dynamic Boards	✓	✓	NO	NO	NO
BG-Bank 256S	✓	✓	✓	✓	✓

GUARANTEED IN YOUR SYSTEM
CROMIX-D • MPM • CCS • OASIS • AMOS

✓ PLUS: 8/16 BIT TRANSFERS • 24-BIT EX. ADDRESSING
8-12 MHZ • 2K DESELECTS • RAM-EPROM MIX
IEEE 696/S-100 • LOW POWER • FULLY STATIC

LITHIUM BATTERY BACKUP avoids power failure crashes intelligently. Unique POWER-FAIL-SENSE circuit allows processor to save register information and disable board before POWER FAILURE CRASHES memory.

BG BANK 256S..... ~~$695~~ Battery Backup ..... ~~$75~~
BG BANK 64S..... ~~$295~~ Battery Backup ..... ~~$50~~

MEGABYTE MADNESS
FOUR 256S CARDS
~~$2295~~ CALL FOR NEW LOWER PRICES

BG COMPUTER APPLICATIONS, 206 Brookside, Bryan, Texas 77801. International orders add 30%.
(409) 775-5009

```
GWW_HINSTANCE
GWW_HWNDPARENT
GWW_HWNDTEXT
GWW_ID
GWL_WNDPROC
GWL_STYLE
```

Note: *GWL_WNDPROC* indicates that a *WndProc* function is specific to a window. The initial value is obtained from the window class but can be changed to make specific instances of a subclass. The possibilities are up to the programmer.

The window's caption text, whose handle is referenced by *GWW_HWNDTEXT*, is easier to manipulate using the following functions.

```
VOID    SetWindowText       ( hWnd, lpString )
WORD    GetWindowText       ( hWnd, lpString,
                              nMaxCount )
WORD    GetWindowTextLength ( hWnd )
```

SetWindowText sets a new caption, and *GetWindowText* copies the existing caption to the specified buffer. The current length of the caption text is obtained by *GetWindowTextLength*.

Window-support functions are numerous, but the following set are generic:

```
BOOL    IsWindow          ( hWnd )
BOOL    IsWindowVisible   ( hWnd )
BOOL    IsIconic          ( hWnd )
BOOL    AnyPopup          ( )
BOOL    EnumWindows       ( lpEnumFunc, lParam )
BOOL    EnumChildWindows  ( hWndParent,
                            lpEnumFunc, lpParam )
```

The first three functions simply allow attributes of a window to be examined. *AnyPopup* is used to see if a pop-up window is displayed on the screen. The *EnumWindows* and *EnumChildWindows* functions are used to let an application get access to the handles of windows that exist in the system. The *lpEnumFunc* parameter should reference a function that receives the window handle as its first parameter and *lpParam* as its second parameter. The function is called with each handle in the respective list. *EnumWindows* searches for tiled, iconic, pop-up, and hidden pop-up windows—in that order.

WINDOW MANIPULATION FUNCTIONS

The following functions are used to manipulate a window in general. These functions usually generate one or more messages to one or more windows that may be affected by changing a window's status or position. These messages are generated by Windows and handled by the appropriate applications.

```
BOOL    ShowWindow  ( hWnd, nCmdShow )
BOOL    OpenIcon    ( hWnd )
BOOL    CloseWindow ( hWnd )
VOID    MoveWindow  ( hWnd, nX, nY, nWidth,
                      nHeight, bRepaint )
VOID    BringWindowToTop ( hWnd )
HANDLE  SetActiveWindow  ( hWnd )
```

ShowWindow sets a window's display mode and position. *NCmdShow* can have one of the following values:

Value	Window Type
HIDE_WINDOW	child or pop-up only
SHOW_OPENWINDOW	
SHOW_ICONWINDOW	tiled only
SHOW_FULLSCREEN	tiled only
SHOW_OPENNOACTIVATE	tiled or pop-up only

NCmdShow can also be a window handle if one window is iconic and the other is displayed. Both must be tiled windows. In this case, the two windows are swapped. A tiled window can be made into an icon if the value is between FF80 and FFFF hex, where the icon position is the seven low-order bits. Likewise, a tiled window can be positioned in the work area at the specified column using values within the range FF40 and FF7F hex. The *OpenIcon* function also performs this operation, but the column is not specified. *CloseWindow* does a *HIDE_WINDOW* operation.

MoveWindow is used for pop-up or tiled windows. The parameters should be self-explanatory. *BringWindowToTop* is used with pop-up or child windows. These types of windows can be layered like stacked paper, where the top position is the top of the stack. Moving a window to the top does not affect the order of the rest of the stack. A window can be forced to be the active window using *SetActiveWindow*. This function is normally used only by Windows itself. The active window is set to null if the specified window does not have the input focus. The return value is the window's handle that previously had the input focus.

WINDOW INPUT FOCUS FUNCTIONS

Only one window can have the input focus of the keyboard and mouse. The following functions are used to force or check the input focus with respect to a window. Normally, Windows handles the input focus based upon mouse movements, button depressions, and pressed keys.

```
HANDLE SetFocus         ( hWnd )
HANDLE GetFocus         ( )
BOOL   EnableWindow     ( hWnd, bEnable )
BOOL   IsWindowEnabled  ( hWnd )
```

SetFocus forces the input focus to a particular window. Null specifies no window. The handle of the previous window with the input focus is returned, and the current value is obtained with the *GetFocus* function.

Although Windows normally handles focus, a window can tell Windows to ignore this selection by using *EnableWindow* with *bEnable* set to false. A window enables selection using a value of true. The current setting is returned by *IsWindowEnabled*.

Regardless of the current *Enable* setting, an application can obtain the current status of a key using *GetKeyState*, as in:

```
WORD GetKeyState ( nVirtKey )
```

The *nVirtKey* parameters are like *VK_LBUTTON* and *VK_HELP*—that is, each key on the keyboard and button on the mouse can be examined. The returned value has the high-order bit set if the key is down. Keys that control toggle flags, such as Caps Lock, have the low-order bit set if the toggle flag is set. Normally, key depressions are fielded as messages to the *WndProc* function of the window with the current input focus.

The next two functions are used to control the way Windows moves the input focus from window to window:

```
HANDLE   SetCapture     ( hWnd )
VOID     ReleaseCapture ( )
```

SetCapture forces Windows to send all mouse movement messages and key or button status messages to the specified window. No other window can be selected by the user, although a *SetCapture* call can be made by any task that is given control of the processor. The *ReleaseCapture* function lets the user regain control of window activation via the mouse or keyboard.

One handy function provided by Windows that is often used with *SetCapture* is automatic generation of notification messages to a window based upon time. An obvious use is to generate a time-out message to a window so an application does not retain control of the mouse and keyboard for too long without some indication from the user.

WINDOW TIMER FUNCTIONS

The following functions are used to control timed events:

```
WORD   SetTimer    ( hWnd, nIDEvent, wElapse,
                     lpTimerFunc )
BOOL   KillTimer   ( hWnd, nIDEvent )
```

SetTimer starts a timed event. The *lpTimerFunc* function is called if the *wElapse* time passes. Note: the function is called at some time after the specified time because another task may have retained control past this time. The time is in milliseconds. The function definition should look like:

```
TimerFunc  ( hWnd, WM_TIMER, nIDEvent, dwTime )
```

where *dwTime* is a long value that is the current system time. *TimerFunc* has the same definition syntax as *WndProc*—the coincidence is intentional. The event identification, *nIDEvent*, can be specified explicitly, or a unique value can be returned by *SetTimer* if the *hWnd* parameter is null.

A time event can be deleted before it occurs using the *KillTimer* function. Killing an event that has already occurred does nothing.

WINDOW SUPPORT FUNCTIONS

Finding windows and some window attributes can be accomplished using the following functions:

```
HANDLE   FindWindow     ( lpClassName,
                          lpWindowName )
HANDLE   GetParent      ( hWnd )
VOID     GetClientRect  ( hWnd, lpRect )
VOID     GetWindowRect  ( hWnd, lpRect )
```

FindWindow returns a nonnull result if a window with the matching class and window name is found. The current parent of a window is obtained using *GetParent*. The location and size of a window's client area and total size can be found using *GetClientRect* and *GetWindowRect*, respectively.

WINDOW CURSOR FUNCTIONS

The mouse position is specified by the cursor, and the keyboard entry position is specified by a caret. The next two sets of functions control these two objects. Note that the two operate independently. First, the cursor functions:

C CODE FOR THE PC

source code, of course

Panache C Program Generator	$150
QC88 C Compiler	$90
EMACS-like Editor	$75
TELE Kernel & Windows	$60
Make	$50
PC/MPX Multitasking Executive	$45
Coder's Prolog	$45
Biggerstaff's System Tools	$40
Translate Rules to C	$30
ICON String Processing Language	$25
Bison (YACC clone) & PREP	$25
LEX	$25
XLT Text Translator	$20

The Austin Code Works
11100 Leafwood Lane
Austin, Texas 78750-3409
(512) 258-0785

Free shipping on prepaid orders MasterCard/VISA

RTC PLUS

TRANSLATE RATFOR TO C . . .
. . . PLUS FORTRAN TO C!

- **VERY COMPLETE TRANSLATIONS OF RATFOR.** Over 95% of STUG's RATFOR is supported!
- **NOW TRANSLATE FORTRAN TO C!** ***RTC Plus*** will now translate your FORTRAN code to C (excluding I/O, complex, and F77 character). Tap vast FORTRAN libraries and accelerate your C development efforts!
- **DEMO IS NOW AVAILABLE** containing restricted versions of the ***RTC Plus*** package.
- **NOW TRANSLATE TWO LANGUAGES FOR THE PRICE OF ONE!**

DEMO: $10
MSDOS: $325

ORDER TODAY!

COBALT BLUE

1683 MILROY, SUITE 101
SAN JOSE, CA 95124
(408) 723-0474

YOUR SYSTEM'S KEY COMPONENT

The Only Magazine By And For Advanced Micro Users.

$3.95
U.S.A.
March/April 1987
Vol. 3 No. 2
Micro/Systems Journal
For the Advanced Computer User
Multitasking Between Programs
see page 22
Also in this issue
Communicating at 9600 bps . . 32
Program Interfacing to Microsoft Windows . . 36
Classic Technology's 286 Speed Pak . . 45
and introducing the column
LANScape . . 76

At last there is a magazine that brings you the strictly technical but practical information you need to stay up-to-date with the ever changing microcomputer technology . . . *Micro/Systems Journal. Micro/Systems Journal* is written with the needs of the systems integrator in mind—the individual who's involved in putting together the hardware and software pieces of the microcomputer puzzle.

In each issue of *Micro/Systems Journal* you'll find such useful and progressive articles as:

- Interfacing to Microsoft Windows
- Unix on the PC
- 80386 Programming
- High Resolution PC Graphics
- Using 80286 Protected Mode
- Multiprocessing and Multitasking

You'll get the hands-on, nuts and bolts information, insight and techniques that *Micro/Systems Journal* is famous for . . . in-depth tutorials, reviews, hints . . . the latest information on computer integration, networks and multi-tasking, languages, and operating systems . . . hard-hitting reviews.

To start your subscription to *Micro/Systems Journal*, simply fill out one of the attached cards or write to *Micro/Systems Journal*, 501 Galveston Dr., Redwood City, CA 94063. You'll receive a full year (6 issues) of *Micro/Systems Journal* for just $20, and enjoy the convenience of having M/SJ delivered to your doorstep each month. Don't wait . . . subscribe today!

```
HANDLE SetCursor     ( hCursor )
VOID   SetCursorPos  ( nX, nY )
VOID   GetCursorPos  ( lpPoint )
VOID   ClipCursor    ( lpRect )
WORD   ShowCursor    ( bShow )
```

The cursor (default is a pointing arrow) is available from Windows or is defined by an application. Application-defined cursors are included using the the resource definition files and are created using the icon editor.

SetCursor is used to change the currently displayed cursor. The previous cursor's handle is returned. The screen position of the cursor is set using *SetCursorPos*. The current position is obtained using *GetCursorPos*.

The cursor moves based upon mouse movements, and it can move over the entire screen. *ClipCursor* is used to limit cursor movement to a smaller, rectangular area. A null parameter resets this area to the entire screen.

The cursor is displayed if the cursor display count is greater than or equal to zero. The *ShowCursor* function increments this value if *bShow* is true and decrements this value if *bShow* is false. The function returns the current value.

WINDOW CARET FUNCTIONS

The caret-control functions are:

```
VOID   CreateCaret       ( hWnd, hBitmap,
                           nWidth, nHeight )
VOID   DestroyCaret      ( )
VOID   HideCaret         ( hWnd )
VOID   ShowCaret         ( hWnd )
VOID   SetCaretPos       ( nX, nY )
VOID   SetCaretBlinkTime ( nMSeconds )
WORD   GetCaretBlinkTime ( )
```

A caret is a vertical line of some sort. Characters are added to the right of the caret, and then the caret is moved to the right of the newly added character. A caret is created using *CreateCaret*, and an application-defined caret is used if *hBitMap* is not null. Otherwise, the caret is a vertical line whose size is specified by *nWidth* and *nHeight*. *DestroyCaret* is used to delete the current caret. There can be only one active caret, and it is associated with a particular window.

The caret can be displayed or hidden using *ShowCaret* and *HideCaret*. These functions operate like the *ShowCursor* function in that a matching number of *ShowCaret* and *HideCaret* function calls toggle the visible state of the caret. A newly created caret is not displayed.

The position of a caret within the corresponding window is set using *SetCaretPos*. There is no *GetCaretPos* because the user does not have the same control over the caret as the mouse. It is up to the controlling application to maintain the caret position and remember where it is.

The blink rate of the caret can be set and returned using the *SetCaretBlinkTime* and *GetCaretBlinkTime* functions.

SYSTEM METRICS FUNCTION

Finally, there are a few general functions that provide access to common Windows attributes:

```
WORD   GetSystemMetrics  ( nIndex )
LONG   GetSysColor       ( nSysColor )
VOID   SetSysColor       ( nChanges, lpSysColor,
                           lpColorValues )
```

The various system attributes are obtained using the *GetSystemMetrics* function. One set of items not accessible by this function is the default system color for various things such as menu bars and caption text. The values of *nIndex* are:

Value	Description
SM_CXSCREEN	screen width
SM_CYSCREEN	screen height
SM_CXVSCROLL	vertical scroll bar arrow width
SM_CYVSCROLL	vertical scroll bar arrow height
SM_CXHSCROLL	horizontal scroll bar arrow width
SM_CYHSCROLL	horizontal scroll bar arrow height
SM_CYCAPTION	caption height
SM_CXBORER	window border height
SM_CXHTHUMB	vertical thumb bar width
SM_CYVTHUMB	vertical thumb bar height
SM_CXICON	icon width
SM_CYICON	icon height
SM_CXCURSOR	cursor width
SM_CYCURSOR	cursor height
SM_CYMENU	single-line menu height
SM_CXFULLSCREEN	full-screen width
SM_CYFULLSCREEN	full-screen height
SM_CYKANJIWINDOW	height of kanji window
SM_CURSORLEVEL	*ShowCursor* level value
SM_DLGFRAME	dialog-box frame width
SM_DEBUG	true if Windows debug version
SM_FULLSCREEN	current full-screen window handle (null if none)
SM_MOUSEPRESENT	true if mouse present

The values of *nSysColor* are:

```
COLOR_ACTIVECAPTION
COLOR_BACKGROUND
COLOR_CAPTIONTEXT
COLOR_INACTIVECAPTION
COLOR_MENU
COLOR_MENUTEXT
COLOR_SCROLLBAR
COLOR_WINDOW
COLOR_WINDOWFRAME
COLOR_WINDOWTEXT
```

The *SetSysColor* function is used to set the currently used colors using two matching pairs of word arrays referenced by *lpSysColor* and *lpColorValues*. The size of the arrays is specified by *nChanges*.

SUMMARY

If you are waiting for a full example program it will be in the next article. So far we have only examined the first part of a Window's application, the functions. Part IV covers the messages that Windows sends to an application's windows. A simple application will then be presented using these functions and messages.

Then is it time to build a real application? Not yet. We still need to see about GDI functions and drawing messages. This will occur in Part V so stay tuned. Then the real fun begins.§

Bill Wong is president of Logic Fusion, Inc., 1333 Moon Dr., Yardley, PA 19067, a systems software development firm.

Buying an AT Clone System

by Sol Libes

To Assemble or Not to Assemble an AT Clone—That is the Question

Prices of AT clones are dropping fast. It is already possible to buy a basic AT clone unit for less than $1,000 and have a complete system for about $2,000. This is easily less than a third of the price of a comparable IBM PC/AT. Not only that, many of the current crop of AT clones offer features not available on the IBM AT. The most common extra features are faster clock speed, more memory on the main board, and higher-capacity hard-disk drives. But there are other additional features as well.

AT clone components are also now readily available. Thus the question becomes, should you buy the individual components and put them together yourself or buy an assembled system?

PUTTING TOGETHER AN AT CLONE

If you are going to assemble an AT clone from the basic components (motherboard, power supply, case, drives, and keyboard), I suggest you refer back to my article on assembling PC/XT clones that appeared in the November/December 1986 issue of *Micro/Systems Journal*. Most of the recommendations in that article also apply to assembling AT clones.

Actually, in many ways it is easier to assemble an AT clone than a PC/XT clone—the reason being that there is more room to work inside the cabinet and the unit assembles with fewer screws. For example, a standard PC/XT power supply is secured with four screws, whereas the standard AT power supply requires only two. The same is true of the motherboard. I would estimate that an inexperienced person could assemble an AT clone in less than an hour, and an experienced person could do it in a third of the time. The modular design of the unit means that the only tools you need are screw and nut drivers.

A typical AT motherboard costs about $500 with 512K of RAM. Most of the boards I have seen are very similar to the IBM AT motherboard—physically and electrically they are directly interchangeable. There are differences, however. The AT clone boards do not include BASIC interpreter ROMs and hence can run only GW BASIC—not IBM BASIC. There are some differences between the two versions of BASIC, and some software cannot run under GW BASIC (this is also true for PC/XT clones).

There are other differences as well between the clone and IBM motherboards. The most important difference is the BIOS. Phoenix Technology has produced a BIOS ROM with a high degree of IBM compatibility. This is probably the most common BIOS ROM supplied with AT clones. There are others, however. The ROM BIOSs from Eden Software, Pyramid Software, and Award Software are also considered quite good.

BUILD OR BUY?

How much do you save by assembling your own AT clone? After checking with several mail-order dealers who sell all the components you would need to assemble an AT clone, I have concluded that you do not save anything by buying components and assembling the basic unit yourself. In fact, you can buy an assembled basic AT clone for less than the cost of putting it together yourself. And there are a lot of disadvantages to buying separate components (refer to my earlier article on assembling PC/XT clones).

My feeling is that, unless you can save a significant amount of money buying components, assembling a unit is a mistake. After checking component prices, I do not understand why anyone would go the assemble-it-yourself route.

BUYING AN ASSEMBLED AT CLONE

At the end of this article, I have provided a list of AT clone vendors that offer assembled units for less than or very close to $1,000. This list is quite long. Generally, these are bare-bones systems from outfits that provide little in the way of support.

For a few hundred dollars more, you can get considerable support and warranty features. Some AT clone suppliers offer a 30-day money-back guarantee if you are not happy with the machine—no questions asked! Generally, these suppliers ask a few hundred dollars more than those without this offer. If you are considering buying a machine and have not been able to find someone who has the same machine to ask them their opinions of it, then you should definitely consider paying for this extra feature.

Another consideration is service. All the low-cost AT clones are available only via mail order. Most come with a one-year guarantee for parts and labor. It still means, however, that if a problem develops, you are going to have to call the ven-

dor, which can develop into a significant phone bill. Or even worse it may require shipping a component or even the entire system back to the vendor, which means that you will not have the system available for use for at least a week and probably much longer.

Some mail-order vendors have arranged for service via a nationwide service organization such as TRW. If local service is important to you, then check that the clone supplier has a local service organization arrangement. This is also worth extra money—don't expect to get this frill on very-low-cost AT clone systems.

The lower-cost systems generally come with only a 90-day warranty. Based on my own experience with four AT clones, I would recommend that you look for a system with a one-year parts and labor warranty. This will generally mean paying more than $1,000 for the basic system, but I think it is worth it.

DISK DRIVES

If you buy an assembled AT clone that includes a hard disk, then be sure to check the performance of the drive. Many suppliers advertise very-low-cost hard-disk drives that have slow access times. The IBM AT uses drives with a typical access time of 38 msec, whereas the IBM XT uses drives with a typical access time of 80 msec. Drives are available with access times of well under 30 msec. Naturally, price is related to speed—the faster the drive, the higher the price. If you are planning to use the system for applications that do a lot of disk accessing, then you will want a fast drive.

The IBM AT is furnished with a quad density (1.2MB) floppy-disk drive as the A drive. The drive can be used with a 360K floppy disk. You cannot format a 360K disk on the drive, however. You can format only 1.2MB disks. Therefore, most users add a 360K drive as a second floppy drive.

Most of the AT clones use the same disk controller card used in the IBM AT—a Western Digital combination floppy/hard-disk controller card. It can handle two hard-disk and two floppy-disk drives. When the controller reads or writes to a floppy-disk drive, it checks to see if it is dealing with a 360K or 1.2MB drive. If you try and use a standard 360K floppy drive with the controller, you will find that the controller will not work with it. Many suppliers sell a special 360K drive for the AT. You can, however, get a standard floppy-disk drive to work by fixing pin 34, on the drive connector, to a high logic level (I do this by putting tape over the connector terminal). The disk controller checks this pin to determine if it is dealing with a 360K or 1.2MB disk drive (1=360K, 0=1.2MB).

WHAT DO YOU HAVE TO ADD?

The assembled systems selling for less than $1,000 are generally lacking components that you will have to add to complete the system. Figure that you will have to add a hard disk (typically about $650 for a 30MB drive with a 38 msec access time), a second floppy-disk drive (figure about $90 if you use a standard floppy), a display controller and display (figure about $200 for a Hercules-type monochrome or about $700 for an EGA combination), and I/O (about $100 for a card that supplies two serial and one parallel port), and don't forget the operating system (MS-DOS will cost you about $70).

Therefore figure that the complete system will set you back between $1,900 and $2,600. A comparable IBM AT system would be between $6,500 and $7,000.

THE KEYBOARD

Generally, AT clones are furnished with a keyboard that follows the old AT keyboard layout. The big difference is that they have only 10 function keys (instead of the 12 on the IBM keyboard). I have yet to see any commercial software that uses the two additional keys and have therefore not found this a problem.

Some of the keyboards have the same spongy feel prevalant on many PC/XT clone keyboards and place the Escape key in strange places. The Maxi-switch keyboard is generally preferred by most AT clone users, it has a good feel and the Escape key is on the left side of the keyboard. Some keyboards have a provision for moving the Escape key from the right to the left side, if desired.

ADDITIONAL FEATURES

Some AT clones have additional features not found on an IBM AT—for example, some include a reset switch. Some system crashes cannot be recovered with the standard Ctrl-Alt-Del key combination and require power to be turned off and turned back on. A reset switch can reset the system without turning off power. This is a worthwhile feature, but it is also a feature that can be added easily and economically to any system. Also, I don't understand why many vendors put this switch on the rear of the unit, making it hard to reach.

A few clones have extra bus slots. The standard arrangement is eight slots on the main board. I have found this adequate for the great majority of users, but if you are going to use the AT for a multiuser application or are planning to plug in a lot of cards, extra slots may be important.

Another feature found on some of the AT clones is a higher-wattage power supply. The typical unit is furnished with a 192-watt power supply (the same as the IBM AT's). Some vendors offer 200-watt, 220-watt, and even 242-watt power supplies—192-watt power supplies are adequate for most systems. A system with several megabytes of memory and a high-capacity hard disk and tape backup unit may need a high-wattage power supply, however.

RAM

Most AT clones are furnished with either 512K or 640K of RAM on the main board. Some include a provision for changing some of the 64-kilobit RAM chips to 256K chips to bring the unit up to 1MB. This additional memory does not conform to the Above-Board specification followed by many software vendors, and it may therefore not be usable with programs such as Lotus 1-2-3. Many disk-caching and RAM-disk programs can work with this linear type addressing system, however.

If you are planning to plug in an extended memory card (for example, an Above-Board memory card), then check that the AT clone will be compatible with it. A few of the AT clones will not work with extended memory cards.

PROCESSOR SPEED

Most AT clones now feature dual-speed operation. The lower-cost units can operate at 6 or 8 MHz. As price increases, the speed options increase—for a little more, you can have a 6/10-MHz unit. Stepping up to 6/12 MHz or 6/16 MHz generally means a big step up in cost. Some copy-protected software can function only at 6 MHz, hence the necessity for this slower speed. Units that have only single-speed operation (typically 8 or 10 MHz) may present a problem with certain software packages. Therefore, being able to switch back to the lower 6 MHz operation is desirable.

Some plug-in cards cannot work at 10 MHz and above. Hence, check out this requirement before buying a high-speed system. The more critical cards are the extended memory cards.

Most dual-speed systems permit changing the CPU's clock speed on the fly. The

method for changing speed is either to use a switch or to type a key combination on the keyboard.

The speed-change switch is usually located on the rear of the unit, although a few manufacturers locate it on the front. There is no doubt that locating it on the front is better. You rarely change the speed, however, so having the switch on the rear is not a serious inconvenience—in some environments it may actually be an advantage.

An alternative method for changing the clock speed is from the keyboard by pressing a special combination of keys (for example, Enter-Ctrl-Alt- + to speed up and Enter-Ctrl-Alt- - to slow down). And one or two units provide some indication of the speed mode that the unit is in. All the units featuring keyboard control start up at the lower speed, and it thus can be annoying to have to always switch the unit to the higher speed after power-up.

As processor speed is increased, particularly to 10, 12 and 16 MHz, you find that software becomes less reliable and often causes the system to hang up. In these cases, you will find yourself changing processor speed frequently and will appreciate having either a front-panel speed-change switch or a keyboard-actuated speed-change system.

Watch out for AT clones that require that you turn the system power off when you change speed. This is a serious inconvenience, and I would avoid such units.

The IBM AT uses a single wait state to match the processor's speed to the slower access time of the RAM and ROM memory. The result is that the processor is not operating at its full speed potential. Most clone systems also operate with a wait state. Some have no wait states, however, and hence operate at full speed. Operating with zero wait states versus one wait state typically provides a 30 percent increase in memory access speed. Note that on a 6/8 MHz machine, this requires RAM chips with 120-nsec or better access times.

Actually, most units contain a jumper or switch on the main board that allows switching the wait state in or out of operation. If the unit you buy has a wait state generating circuit, look for the jumper or switch that turns it on or off. Then you might try disabling the wait state circuit to see if the system can operate with it removed. Frequently, units come with faster memory than was available when the unit was originally engineered, and the system operates reliably without the wait state delay. If you find that the system sometimes hangs up for an unexpected reason, put the wait state back in by replacing the jumper or setting the switch.

I/O

Most of the lower-cost units do not include any serial or parallel ports, which means you will have to add an I/O card to provide them. It is possible to add two serial ports, a parallel port, and game port with one plug-in card. Some AT clones include two serial ports and a parallel port on the main board. This approach frees up one bus slot. If you plan on installing a large number of plug-in cards, this feature may also be important to you.

IBM, on its serial ports, has changed from the old familiar 25-pin DB-type connector to a 9-pin DB-type connector. It takes up less space, costs less, and actually makes more sense because most serial devices use only nine lines or less. Some clone makers have followed in IBM's footsteps and are also furnishing 9-pin serial ports. This means that if you are using serial devices (for example, a modem or serial printer), you will need to make up, or buy, a 9-pin to 25-pin adapter.

SETTING UP AN AT CLONE

Hard-disk drives, as shipped by the vendor, have been formatted at a "low level." It is therefore necessary to do a high-level format of the drive and to set certain system parameters. IBM provides an Advanced Diagnostics disk with software to do this and also to test the hardware. Some AT clone makers have put these setup and diagnostic routines in ROM. You can invoke them by typing a special key combination (for example, Ctrl-Alt-Enter). Some AT clone makers provide a diagnostic setup disk and instructions on its use; some AT clone makers provide neither. Therefore, be sure to check with the vendor that this software is provided in ROM or on disk.

If you do need the AT setup and diagnostic software, note that this software exists in the public domain. There are also some excellent disk-test utilities that are superior to those provided by IBM. You will have to check around on some electronic bulletin-board systems to find them.

I recommend that after you buy an AT clone system with a hard-disk drive and after you set it up, you do a test of the hard disk to be sure there are no problems. Run the test several times.

One other problem—hard disks have been known to develop problems that require a low-level reformat of the disk (you loose all data when you do this). On the XT you do this by accessing a routine located in the ROM on the hard-disk controller. The AT's disk controller does not contain such a routine, however. You can find some low-level format routines in the public domain (check some local bulletin boards).

While on the subject of setup, note also that you need the setup program to set up the clock/calendar circuit. The system configuration data is maintained in a CMOS RAM chip on the main board. Battery backup is provided to retain this data. On the IBM AT a Nicad battery is provided for this RAM and the clock/calendar registers (Nicad batteries have a typical life of 5 years). Some AT clones use standard batteries for this purpose, which I find an inconvenience. Standard batteries are good for only a year and a half at best. When the battery maintaining the system data runs down, the system will not operate properly, so I prefer a system with a Nicad battery.

CONCLUSION

AT clones come with a wide variation in features. The purpose of this article is to point out these different features so that, when you call an AT clone vendor, you can ask about the features that the unit you are interested in has or doesn't have.

Already, more than three dozen different AT clone systems are available to compete with the IBM AT. Although some are very close copies of the IBM machine, most offer additional features to those of the IBM system.

The prices of AT clones range from virtually the same as the IBM AT (for example, the Hewlett-Packard and Compaq systems) to prices that are less than one third the price of the IBM AT. The list shown in the sidebar contains 13 vendors of system in the $1,000 and less class (assembled basic configuration).

I would be interested in hearing from readers who have purchased or assembled AT clones about their experiences with them. §

LOW-COST AT CLONES

Several AT clones sell for less than $1,000 or very close to it. The following is a list of the ones I am aware of. Unless otherwise indicated, the systems run at 6/8 MHz and do not include any drives, controllers, RAM, or I/O. From the listing, it looks as though California is the AT clone leader of the country.

LOW-COST AT CLONE VENDORS

Bentley Computer Products
(800) 626-4027
(512) 250-9897
$995
Includes 512K RAM, 1.2MB floppy, and dual disk controller.

Club AT Inc.
Fremont, CA
(415) 490-2201
$755

Compuworld
Sunnyvale, CA
(408) 749-9935
$725

Data Bank Computer
Fremont, CA
(415) 490-5002 (sales)
(415) 490-5135 (tech assistance)
$849 6/10 MHz
$795, quantity 3

Focus Technology
Irvine, CA
(800) 843-5789
$559, mini-case
$699, full-size case
$788, 6/10 MHz
Quantity 3 prices

Forton Corp.
Fremont, CA
(415) 490-8171
$745, quantity 2, 640K RAM
$785, 6/10 MHz
$1,185, 6/12 MHz

Intelligent Micro Systems
San Antonio, TX
(800) 624-4031 (sales)
(512) 250-8271 (tech assistance)
$1,035, 6/10 MHz
Includes 512K RAM, 1.2MB floppy, and dual disk controller.

Mctek Systems Inc.
Berkeley, CA
(415) 843-0714
$799, 6/10 MHz

PC Source
Austin, TX
(800) 643-0992 (sales)
(800) 527-3125/
(512) 331-6700 (tech assistance)
$995
Includes 512K RAM, 1.2MB floppy, and dual disk controller.

PFM Computers
Sausalito, CA
(415) 331-1986
$1,695
Includes 20MB hard disk and monochrome adapter/display and 512K RAM. Take away the extras, and the basic system works out to be less than $1,000.

SF Micro
San Francisco, CA
(800) 237-5631
(415) 751-7781
$759

Southern California Systems Inc.
Los Alamitos, CA
(800) 237-7274/
(213) 493-6633 (sales)
(213) 596-9748
$1,028
Includes 1.2MB floppy and dual disk controller.

Supreme Co.
Milpitas, CA
(408) 262-6888
$825
$775, quantity 3

$99 BUSINESS BBS

24 hour business information center via modem (300-2400 baud). Setup custom multi-level menus, data entry forms and info-sheets easily with no programming req'd. Integrated data base, XMODEM up/downloads, remote PC operation. With source code $249.

$99 PC COMMX $119 CP/M

Emulates: VT100/102,Wyse,HP,ADM,TV,IBM,ADDS Transfers: KERMIT, XMODEM, COMMX mainframe, TLX/TWX. POPUP hotkey to DOS or programs. Unattended control scripts, dial directory for 700 entries & electronic mail features.

$59 C DATA ENCRYPTION

Data Encryption Standard (U.S. government standard FIPS PUB46) in Microsoft "C". Includes compression & telecomm formatting, allowing faster transmission & less storage space plus compatibility with any computer or service. Complete "C" source code $249.

HAWKEYE GRAFIX Inc
BOX 1400, OLDSMAR
FLORIDA 33557
DIAL 813-855-5846

386•386•386•

Asking Questions and Getting Responses in an MS-DOS Batch File

by Bill Rogers

With a capability to ask questions and get responses in an MS-DOS batch file, you could take alternate actions and construct simple menus using the batch file. Unfortunately, no MS-DOS batch mechanism exists to do this. You can, however, write a program to supply this capability. The program could ask the question and supply the response to the batch file either by setting the ERROR LEVEL code or by setting an environment variable. You could then test the ERROR LEVEL or environment variable by means of a batch file command.

The program QRYRSP (Listing 1) sets the ERROR LEVEL code. The command line for QRYRSP is:

```
qryrsp{⟨responses⟩{⟨query⟩}}
```

Here, a response is a single character; the ERROR LEVEL code set corresponds to the index of the user's single-character response in the <responses> string; and <query> defines the query, which may be more than one word. The default for <responses> is 0123456789, which is translated to the same ERROR LEVEL code, and the default for <query> is "Enter Character." The string ": _" is automatically supplied after the query.

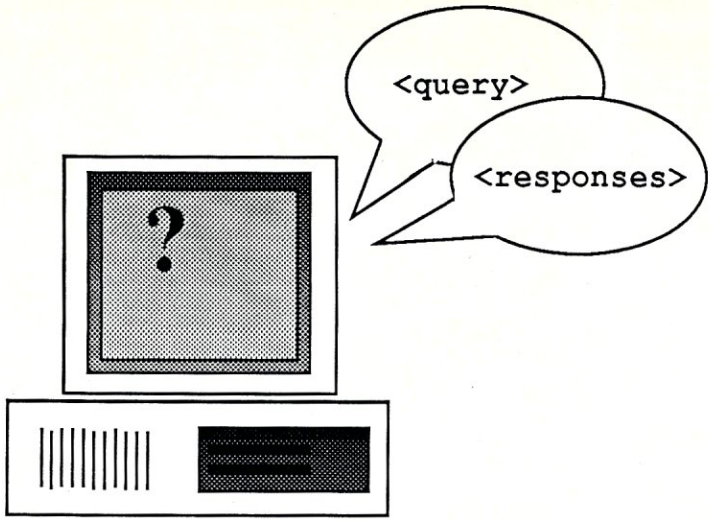
The batch file DEMOQR.BAT (Listing 2) demonstrates the use of QRYRSP, and the console output from this demonstration is in Listing 3.

The program QRYRSP is written in C. It was compiled using Computer Innovations C86, Version 2.30f, and the demonstration was run using MS-DOS, Version 2.11, on a Lomas Lightning (8086) system. §

Bill Rogers is a consultant in Princeton, New Jersey.

Listing 1

```
/*  FILE NAME:                  qryrsp.c
 *  FILE DESCRIPTION:           Query for single character.
 *  SOURCE LANGUAGE:            C86/2.30f.
 *  AUTHOR:                     Bill Rogers.
 *  CHANGE(VER,DATE,TIME,NAME): 5.0.2  86/11/23  22:00:00  Bill Rogers
 *  GENERAL DESCRIPTION:
 *      Query for Single Character and Translate to Exit Value (i.e.,
 *      Error Level).
 *
 *      the calling sequence is
 *
 *          qryrsp {<argument 1> {<argument 2>}}
 *
 *      <argument 1> gives the allowable one-character responses as a
 *      string. The translation rules from one-character response to
 *      Exit Value (i.e., Error Level) are:
 *
 *          Char Posn In    Error
 *          <argument 1>    Level
 *          ------------    -----
 *              0            0
 *              1            1
 *              2            2
 *              ...          ...
 *              n            n
 *
 *      Only those characters given in <argument 1> will be accepted.
 *      However, if no <argument 1> is given, the argument defaults to
 *      "0123456789".
 *
 *      <argument 2> gives the query. However, if no <argument 2> is
 *      given, the argument defaults to "Enter Character". A colon,
 *      space, underline and backspace after the query are
 *      automatically supplied.
 *
 *  SCCS IDENTIFICATION */
        static char SCCSID[]="@(#)qryrsp.c  5.0.2";
/*---------------------------------------------------------------------*/
/*  INCLUDE */

#include    <stdio.h>
/*---------------------------------------------------------------------*/
/*  DEFINE */

#define     local       static  /*  not external */
#define     metachar    int     /*  integer representation of char */
#define     public              /*  declaration of external */
#define     string      char    /*  character array that is string */
#define     true        1       /*  boolean true */
#define     void        int     /*  function returns no value */
/*---------------------------------------------------------------------*/
/*  CONSTANTS */
```




```
local string    DFLTQRY[] = "Enter Character";  /* default query */
local string    DFLTRSP[] = "0123456789";       /* default responses */
/*-------------------------------------------------------------------------*/
/*  VARIABLES */

local metachar  C;              /*  response to query */
local FILE     *CONSHNDL;       /*  console "handle" file */
local int       ERLVL;          /*  ERRORLEVEL */
local char     *ERLVLPTR;       /*  pointer to char in response
                                    string */
local int       I;              /*  index */
local string    QRY[81];        /*  query */
local string    RSP[81];        /*  response string */

extern metachar fgetc();
/*-------------------------------------------------------------------------*/
public void main(ARGC,ARGV)
    int     ARGC;       /*  in */
    string *ARGV[];     /*  in */
{
    static string   FUNC[] = "main";

/* begin */

    if        (ARGC == 1) {

     strcpy(RSP,DFLTRSP);
     strcpy(QRY,DFLTQRY);
 } else if (ARGC == 2) {
     strcpy(RSP,ARGV[1]);
     strcpy(QRY,DFLTQRY);
 } else if (ARGC >  2) {
     strcpy(RSP,ARGV[1]);
     QRY[0] = '\0';
     for (I = 2 ; I < ARGC ; I++) {
         if ( I > 2) {
             strcat(QRY," ");
         }
         strcat(QRY,ARGV[I]);
     }
 }
 CONSHNDL = fopen("CON","rb");
          /*  open console file in "raw" mode */
 if (CONSHNDL == NULL) {
     fprintf(stderr,"\n**** Console File
           Did Not Open ****\n");
     exit(9999);
 }
 printf("%s: _",QRY);
 while (true) {
     printf("\b_\b");
     C = fgetc(CONSHNDL);
     fputc(C,stdout);
     ERLVLPTR = strchr(RSP,C);
```


#1 C interpreter

C-terp

The professional C development environment

Your C compiler creates great final code . . . but as a programming tool, it's too, t o o s l o w.

With C-terp you can edit, debug, and run without the wait. Nothing, but nothing, is faster for developing professional C programs.

Choose the perfect C-terp companion for your C compiler

C-terp/Microsoft	C-terp/XENIX
C-terp/Lattice	C-terp/Aztec
C-terp/Mark Williams	C-terp/C86

Link in all your compiler's functions, your own functions, add-on libraries, assembly routines, and data objects. Get instant access to everything in the C-terp interactive environment.

Only C-terp offers all this and more

- **Full K&R with common ANSI enhancements**
- **Source level interactive debugging**
- **Software paging for your big jobs**
- **Complete multi-module support**
- **Run-time pointer checking**
- **Unsurpassed reconfigurable screen editor**
- **Dual display and full graphics support**
- **Large model** ■ **Call-in**

ORDER C-terp TODAY (specify compiler)

C-terp runs on IBM PC, AT or compatibles.

Price:
MS-DOS 2.x and up - $298,
Xenix System V 286 - $498
MC, VISA, COD
30-day money-back
GUARANTEE

Trademarks: C-terp (Gimpel Software). C86 (Computer Innovations). Lattice(Lattice, Inc.). Xenix, Microsoft, MS-DOS (Microsoft, Inc.). Aztec (Manx Software). Mark Williams (Mark Williams Company). IBM (International Business Machines, Inc.)

GIMPEL SOFTWARE

3207 Hogarth Lane, Collegeville, PA 19426

(215) 584-4261

B-R-E-A-K THE DIRECTORY BARRIER WITH DPATH + PLUS™

- Run WordStar, dBase, 1-2-3 or any other program from any disk directory, without having to copy overlay files, help files, dictionaries, libraries and the like into that directory.
- Find **ANY** file for **ANY** program, even if the program wants its files to reside in the current directory.
- **Output** files can even be created in other directories.
- Assign directories to pseudo-drives, such as F:, and refer to the directory using only the pseudo-drive id.
- **Protect** your hard disk from accidental formatting (true IBM compatibles only).
- DPATH+Plus specs are updatable at any time, even while a program is running!
- Can be deactivated and later reactivated.
- A **trace**-mode facility allows you to see what file-oriented calls any program makes to DOS.
- Operation is completely transparent to your program. **Fast** — written in assembly language.
- Works with hard disks, floppies or RAM disks, in any combination. Requires DOS 2.0 or later. Runs in 4-9 KB of memory, depending on features used.
- **Not** copy-protected. Can be installed on hard disk.

$45 + $5 s/h.
COD add $3.
Visa/MC/check.

Personal Business Solutions, Inc
PO Box 739, Dept P
Frederick MD 21701
☎ 301-865-3376

60-day money back guarantee!

THE PROVEN MICRO FAMILY for all the INDUSTRY APPLICATIONS

SLICER®

*Specials **New Low Prices!**

*Multi-User Slicer Combo Slicer SBC A&T (No RAM) Slicer 1 MEG Board A&T (Complete) CCP/M by Digital Research Inc.	$950.00
Slicer SBC A&T 256K	$445.00
*Slicer SBC A&T 128K with double deck sockets for additional 128K	$350.00
*Slicer SBC A&T (No RAM)	$295.00
*Slicer 1 MEG Board A&T	$450.00
*Slicer 1 MEG Board (Full Kit)	$295.00
Slicer PC Expansion Board A&T	$495.00
CCP/M (Digital Research Inc.)	$250.00
MS DOS (Micro-Soft Corp.)	$175.00
*New Slicer Bios for MS DOS	Call

Other kit forms available - Enclosures & other support hardware in stock - Call or write for latest information & prices!

SLICER COMPUTERS INC.
3450 Snelling Avenue S.
Minneapolis, MN 55406
(612) 724-2710


```
        if (ERLVLPTR != NULL) {
            ERLVL = (int)(ERLVLPTR - RSP);
            break;
        }
    }
    printf("\n");
    fclose(CONSHNDL);
    exit(ERLVL);
} /* main */
```

Listing 2

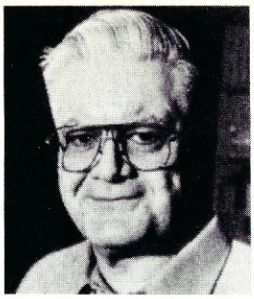
```
:   demoqr.bat   1.0.1  86/09/10  11:00:00  Bill Rogers
echo    off
echo
echo Choose Your Fruit:
echo
echo     a) Apple
echo     b) Banana
echo     c) Cherry
echo     p) Peach
echo
qryrsp abcp Enter Choice
if      ERRORLEVEL 3 goto peach
if      ERRORLEVEL 2 goto cherry
if      ERRORLEVEL 1 goto banana
if      ERRORLEVEL 0 goto apple
:apple
echo
echo Apple Chosen
goto    end
:banana
echo
echo Banana Chosen
goto    end
:cherry
echo
echo Cherry Chosen
goto    end
:peach
echo
echo Peach Chosen
goto    end
:end
```

Listing 3

```
D:\CLIENT\APPLICAT\LANG\BIN > demoqr

D:\CLIENT\APPLICAT\LANG\BIN > echo    off

Choose Your Fruit:

     a) Apple
     b) Banana
     c) Cherry
     d) Peach

Enter Choice: c

Cherry Chosen

D:\CLIENT\APPLICAT\LANG\BIN >
  a) Apple
     b) Banana
     c) Cherry
     d) Peach

Enter Choice: c

Cherry Chosen
```

The Scientific Computer User

A.G.W. Cameron



Number Crunching on a Sun 3/260 Workstation

WHY A SUN WORKSTATION?

I mentioned in the January/February 1987 issue of *M/SJ* that I was awaiting delivery of a Sun 3/260 Workstation. Those words were written around the beginning of November 1986. It arrived, a little earlier than expected (thanks to the fact that I beat most of the rush by ordering it right after the August announcement) about the middle of November. Now, two months later, I can report on a mixture of elation and frustration. I will get to these in due course.

My decision to get a scientific workstation occurred in March 1986. I had been engaged with two colleagues, Willy Benz and Wayne Slattery, in a long series of three-dimensional hydrodynamic calculations using Cray XMP supercomputers at Los Alamos. We had used literally hundreds of hours of time on these machines. Willy and I were at a meeting in Houston giving the results of one of the projects—the investigation of the largest collision the planet Earth was likely ever to have encountered, a collision with a body larger than the planet Mars, with the postulated outcome that the collision lead to the formation of the Moon. We had found conditions in which that did happen but not quite in the way we had originally postulated.

Willy was a young postdoctoral fellow at Los Alamos, having come from Switzerland, and the 3D hydrocode was his. During the Houston meeting he told me that he was interested in coming to Harvard for a year and using money from his fellowship from the Swiss National Science Foundation to support himself. I told him that he would be most welcome and that I would see what was the best computer I could get with the salary money I would save by not having to support him so that we could continue doing hydrodynamic calculations at a necessarily reduced speed. Government grants for research operate in a somewhat peculiar way: when you hire someone you must pay for benefits on his salary and then you must pay university overhead on top of that, so in the end you generally get charged about twice the amount of the salary from the grant. On the other hand, when you buy equipment, no benefits or overhead are charged, so equipment money goes twice as far as people money. This meant I would be looking for the best number-crunching computer I could buy in the $40,000 to $50,000 range.

This is a range in which many different scientific workstations are available, often with substantial educational discounts. One class of workstations, the MicroVAX II, uses proprietary CPU chips, but just about all the others use the Motorola 68020. There is one other proprietary system, the IBM RT/PC, that extends into that price range, but it is so much slower than all the others that a number-crunching scientist would be crazy to consider it.

The Motorola 68020 is usually paired with the 68881 floating-point coprocessor. In the past I have usually given the performance of PCs in terms of fractions of the floating-point performance of the DEC (Digital Equipment Corporation) VAX 11/780. Continuing that practice for now, experience has shown that a MicroVAX II performs at about 0.7-0.9 of the VAX 11/780. Systems containing a 68020 running at 16.67 MHz with a 68881 running at 12.5 MHz generally slightly underperform the MicroVAX II. This would also have been true of the Sun Microsystems 3/160 in its basic configuration.

The Sun 3/160, however, also had a floating-point accelerator (FPA) option that made the 3/160, so equipped, stand out well above the competition. This option board used the Weitek 1164 and 1165 floating-point chip set. I first mentioned these chips in my article in the November/December 1985 issue of *M/SJ*, at which time I expressed the hope that these chips would soon be made available for PCs as floating-point coprocessors. This has yet to happen, but they have now appeared in that role in the Sun FPA. Sun's benchmarks indicated that the FPA outperformed the VAX 11/780 by about a factor of three for floating-point operations. Once that was established, I never had any real doubt that I had to get a Sun workstation that contained an FPA.

I was not ready to place an order right away, however. First, I had to get permission from my granting agency (NASA) to change my budgets to allow an equipment purchase. It was June before that paperwork was completed. By that time I had become aware of unannounced developments at Sun Microsystems. There was going to be a new desktop workstation that could take the FPA. There was also going to be a successor to the 3/160 that would use a higher clock speed and cache memory and that would run much faster than the 160. Some of this would be announced in August. So I held off to see what opportunities the announcement would bring. It turned out that all of the above features were announced. The new desktop machine became the 3/110, which had three slots. This was not enough for me because my bare minimum needs were for a CPU board, one or more extra memory boards, a hard-disk controller, and the FPA. The successor to the 3/160 was the 3/260. In the 260 the 68020 ran about twice as fast as in the 160 through the combination of a faster clock (25 MHz) and the cache memory. The FPA ran about 40 percent faster primarily because the cache reduced the time for transferring data and results to and from the FPA board, indicating that the speed of the FPA had been bus-limited.

Thus it was that a few days after the Sun announcement I ordered a 3/260. I took a package deal in which the 25-MHz 68020 and the 20-MHz 68881 were combined with 8 megabytes of error-correct-

ing memory (120-nanosecond dynamic chips), one 280-megabyte hard disk with an SMD interface, a 60-megabyte tape cartridge drive with a SCSI interface, and a new high-resolution landscape monitor with 1,600 × 1,280-pixel resolution. I also ordered the FPA and an interface kit that would allow me to run my Apple LaserWriter from the Sun. What brought all this within my price range was that we became eligible for a 30 percent discount by adhering to a technology transfer agreement worked out with Sun by the Space Telescope Science Institute. Sun bundled a lot of software into the package: C, Pascal, FORTRAN, and a lot of Sun extensions to UNIX.

EARLY FRUSTRATIONS

Things started going wrong right at the beginning. When the truck arrived containing the parts of the system, we had to enlist the help of two hefty volunteers from the Center for Astrophysics mailroom to get the pieces off the truck and into my office. The CPU and its associated boards were in one free-standing pedestal and the 280-megabyte drive was in another similar pedestal, and both of them were *heavy*. The truck was not a Sun truck; it was from a shipping company. After it left we found there had been a mistake; one of the pieces was a Sun 3/50 workstation that did not belong to us and the piece that had all the documentation was missing.

A couple of days later, two Sun field engineers appeared to install the system. This they did to the best of their ability, but this was the first time either of them had seen a 260, so they were flying slightly blind. They brought the system up, but they did not install the FPA correctly. I did not discover this until we had documentation and tried to use the FPA.

The replacement documentation took two weeks to arrive. This was an early indication to me that Sun Microsystems had developed a bureaucracy in the process of growing rapidly. Two important pieces of documentation never came—the hardware-installation manual and the FPA manuals. I managed to get the latter on a later occasion from a Sun field engineer who happened to have a set in his car. I still do not have the hardware-installation manual, even though I have complained bitterly to every Sun employee I have contacted for any reason, usually getting a response that it would be reported.

The Center for Astrophysics has an Ethernet cable going around its complex of buildings. This is ideally matched to the needs of Sun workstations, which connect to one another via Ethernet. Sun has developed a Network File System (NFS) software package that allows certain Sun workstations to be equipped with large disks and designated as file servers. The client Sun workstations boot off this disk via Ethernet and read/write to and from the server disk over the network; they are thus known as diskless workstations. Sun has also developed a PC version of this program—the PCNFS—that works with 3Com Ethernet cards for PCs, and I ordered one to communicate with the Sun. My original intent was to connect the Sun, two PCs, and a diskless 3/110 workstation being ordered by a colleague, which will run off my disk, all to the Ethernet cable outside my office through a multiple-port transceiver called a Delni, manufactured by DEC. As a first step in that direction, our central computing facility lent me a Delni and the thick Ethernet cables to connect my XT clone to the Sun.

I then installed the software as best I could on both the Sun and the PC, somewhat hampered because this was my first experience with UNIX and I was having to learn everything as I went along, which was not easy because most Sun documentation was cryptic. I established a very shaky communication between the two computers. I was able to transfer somewhat large files between them in each direction but very slowly. It was clear that the computers had to retry the sending of each packet many times before the other would accept it. Thus, a 300K file took 10 to 20 minutes to transfer and was likely to time-out about four times along the way, requiring me to tell it to retry. The importance of this marginal success was that we were able to get the source code for the 3D hydrocode into the Sun and thus could start to work with it. This source code contained nearly 10,000 lines of FORTRAN.

When you phone the Sun help number, a recording starts playing that tells you to be prepared with your model and serial number and a purchase-order number or a credit card unless you are within the warranty period. It then tells you to press one button for software help, another for hardware help, and yet a third for a new installation. I thought my problem with the Ethernet connection was probably connected with software, so I chose to go that route. My problem was assigned a service-order number, and the person who took note of the problem said that the person who was familiar with Ethernet software had gone home and that she would talk the problem over with her in the morning and get back to me. The "get back to me" never took place, even though a month went by and I enquired about the status of the service-order several times, always to be told that Sun was working on it. Finally, my Sun salesman came around, thought about the problem, concluded that it was probably a hardware problem, and suggested that I call to have the service-order reassigned to the Boston-area hardware people. This I did and was soon visited by the people who had installed the system. They made some calls, then told me that the problem was a known one and that the ports I was using on the Delni would have to be upgraded to a new revision level. Our computer-center management was somewhat skeptical of this finger pointing because other 3Com PC cards were working OK with other Delnis. This is where this matter stands at the moment.

As soon as we started compiling the hydrocode, we discovered the FPA installation problem. A Sun field engineer was able to talk me through the correct installation procedure over the telephone, so fortunately that was a short-lived problem.

With these various frustrations, I have not yet tried to bring up any of the more advanced Sun workstation features, such as the windows environment or the various graphics packages. We have concentrated on number crunching. Even there there is a residual problem. We cannot compile and link the hydrocode with optimized FORTRAN, because we run out of space in the linking procedure. We can sometimes, but not always, use partial optimization, the failures again being because we run out of space. It is clear that I will have to remake the disk partitions, allowing more space for swapping and such things. I am waiting to do that until I get more experience with backing up the disk and until the Ethernet problem is resolved.

NUMBER CRUNCHING

There is a more cheerful side to the story. It concerns the use of the Sun as a number-crunching tool. To explain that I must tell you something about the 3D hydrocode.

Hydrodynamic calculations determine the flow of a fluid through a geometric construction of some kind. Usually, you can think of the fluid as occupying cells in a mesh. If the cells move together with the fluid contained within them, this is called Lagrangian motion. This type of motion is easily treated in one dimension, where that

dimension may be a radial distance in an object that is spherically symmetric (like a star) or a distance down a pipe for the case of linear flow. If the cells are fixed in space and the fluid moves past them, this is called Eulerian motion. This is usually what must be used for flows in two- and three-dimensional geometries. A good rule of thumb is that it is desirable to have something like 100 mesh points for each dimension in the problem. This means 100 points for a one-dimensional problem, 10,000 points for two dimensions, and 1,000,000 points for three dimensions. That is why supercomputers have usually been needed to do three-dimensional hydrodynamic calculations.

Willy Benz's 3D hydrocode does not use cells and mesh points. It is called a smoothed particle hydrodynamics code. The matter that is present in space in a problem is divided into discrete particles, and these are free to move in three dimensions. The particles are defined to have finite extensions in space that strongly overlap at normal densities. Fluid properties such as pressure and density are determined by doing local averages over the neighborhood of a particle. Willy has found that quite satisfactory resolution is obtained in such problems as the collision of planets by using about 4,000 particles. This makes the problem of three-dimensional motions much more tractable, even on large computers.

Even so, when these particles are attracted to each other by mutual gravitation, the problem is still immense. In each time step, it is necessary to compute the distances between each pair of particles in order to obtain the gravitational forces on each particle. This is an example of an *n*-body problem, in which the number of calculations required varies as n^2. Thus, with 4,000 particles, nearly 8,000,000 distances need to be computed in each time step. This is what was being done with Willy's code on the Los Alamos Crays. These distance calculations were "highly vectorized," which means that they were efficiently done with a great deal of parallelism on the Crays.

About the time that Willy arrived at Harvard, at the end of September, my colleague Bill Press had developed some new algorithms for use with *n*-body problems. You have met Bill in these pages before; he is the senior author of the Cambridge University Press book *Numerical Recipes*, which I reviewed in the July/August 1986 issue of *M/SJ*. Bill developed a "tree code" for handling such problems. In this scheme a tree is developed that has the individual particles as nodes at the lowest level. The next level of nodes is obtained by combining mutual nearest neighbors and taking the center of mass and quadrupole moment.

This process of combining mutual nearest neighbors continues until, at the highest level, there is just one node containing all the particles. In the process of obtaining the gravitational forces on a particle, the tree is "opened up" only as much as is necessary; distant nodes need not be opened up in order to determine their gravitational forces on the particle. The degree to which the tree is opened up depends upon the desired accuracy of the problem.

The *n*-body problem with the tree code becomes an *n*log*n* problem, in which the amount of computation does not increase so rapidly with increasing *n*. Nevertheless, there is a lot of overhead involved in the tree calculation. Furthermore, the tree code procedure cannot be vectorized. This gives the Cray XMP a large penalty. The tree code on the Cray becomes faster than the straightforward procedure for *n* greater than almost 2,000, which means that we gained about a factor of 3 speedup in running the 3D hydrocode using the tree. This would be worthwhile in its own right. Consider the effect on the Sun, however. Then the tree code wins for *n* greater than about 200.

Single precision on the Cray is the same as double precision on the Sun. We have found that for the 3D hydrocode on the Sun in single precision and with unoptimized FORTRAN, we run at 10 percent of the speed of the Cray XMP using the tree code. We lose by a factor of 1.4 in going to double precision on the Sun. We gain a factor of 1.4 in going to partial optimization with the FORTRAN. Thus, with everything in our favor, we can run at about 14 percent of the speed of the Cray. Because the Sun is dedicated to this problem, at least currently, the net result is that we can solve problems faster on the Sun than on the Cray because we have in general only been able to get about one to two hours per night on the Cray.

That is the real measure of the power of the Weitek chips.

THE WEITEK CHIPS

The Weitek chips do 64-bit arithmetic, less precise than the internal 80-bit arithmetic on the 8087. They do not do all the operations in the IEEE standard; they do the simple arithmetic operations and they move numbers between registers, but they do not do trigonometric functions or exponentiation or take logarithms. When one of these operations is called for in the Sun FPA, the board detects an exception. The operation causing the exception is then performed on the 68881 on the CPU board. This is a clever trick that maintains the IEEE standard without a significant effect on the throughput for an average operations mix.

A little more than a year ago, I was told by an expert in chip design that the current state of the art would allow a floating-point chip set to be designed that would go about four times as fast as the Weitek chip set. It is interesting in this regard to wonder where the next big speed increase in Sun workstations will arise. Sun has announced as its policy that it will try to double the workstation speed every year. Its next announcement will be due presumably in late summer. I understand that prototypes of its next machines are being tested. It is easy to guess that the CPU will be the 68030. But the floating-point coprocessor? A big increase in the clock speed in the Weitek chips? A new generation of Weitek chips? Or new chips from someone else? It will be an interesting announcement.

It is also interesting to speculate on how rapidly the workstation capability of the 3/260 will become available on new machines. The Apple "open architecture" Macintosh II has just been announced as this article is in proof. The machine uses 68020 and 68881 chips running at just under 16 MHz. That gives it a performance comparable to the basic Sun 3/160 without FPA. It is reasonable to expect that third-party developers will produce FPA boards aimed at the Macintosh that will use the Weitek chip set. Since the Macintosh II sells for much less than $10,000, you may soon be able to buy a machine three times faster than a VAX 11/780 for less than $14,000.

At that point the Crays will mainly be useful for those who need to do problems that require thousands of hours of Cray time each year. The average Cray user will probably be better off with a personal workstation using a Weitek chip set. §

ADDRESS

Sun Microsystems Inc., 2550 Garcia Ave., Mountain View, CA 94043; (415) 960-1300.

A. G. W. Cameron is Professor of Astronomy at the Harvard-Smithsonian Center for Astrophysics, 60 Garden Street, Cambridge, MA 02138.

The UNIX File

Ian F. Darwin

A UNIX-Like System With Full Source and Build Your Own AT&T 3B Computer From an SBC

This column discusses the UNIX operating system. If you have comments or questions about UNIX or this column, please write to Ian Darwin at Box 603, Station F, Toronto, Ontario, Canada M4Y 2L8. If you have UNIX mail access to the uucp network, mail "ihnp4!darwin!ian." Internet users should try "ian@darwin.uucp." I can't always answer immediately, but I will get back to you; electronic mail gets answered first!

This month I'll start by looking at a book (and source tape) for PC owners interested in UNIX. I'll also discuss material for well-endowed hackers or small-system integrators who want to get started with VME systems (the VMEbus was described in the January/February 1987 issue of *Micro/Systems Journal*).

UNIX-LIKE SYSTEM WITH FULL SOURCE

Andy Tanenbaum, author of the well-known networking text *Computer Networks: Toward Distributed Processing Systems* (Prentice-Hall 1981), has turned his attention to operating systems. As is the practice, he has written a UNIX-like system called MINIX (minimal UNIX) and a textbook that go together for the study of operating systems. The new book, *Operating Systems: Design and Implementation* (Prentice-Hall 1987, ISBN 0-13-637406-9) is heavy reading at 719 pages. It talks about operating systems in general and Tanenbaum's own in particular. In addition to more than 400 pages of textual discussion of operating systems, the book contains the manual pages for MINIX, a source-code listing and cross-reference of the operating system (kernel only, not including bootstrap or utilities—253 pages), and an appendix describing how to recompile the system from the sources supplied. In keeping with the trend in UNIX books, the publishers have made the source code for the system available in machine-readable form.

Unlike some other demonstration operating systems, MINIX was intentionally patterned after the external appearance of V7 UNIX. Tanenbaum has written code that is similar in function to the V7 UNIX system. His MINIX system does not contain any AT&T code, so it can be distributed with source code. And unlike some other UNIX clones for which source may eventually become available, MINIX was designed to fit within the constraints of the IBM PC. The distribution includes binaries for the IBM PC, PC/XT, and PC/AT and 100 percent-compatible clones.

My review copy of the MINIX software did not arrive in time for detailed analysis, although I do have a copy of the textbook. And I have seen MINIX running—Prentice-Hall was demonstrating MINIX on a PC/AT at the Washington UniForum trade show in January. Tanenbaum provided the following list of features and programs:

- system-call compatible with V7 UNIX (several minor calls missing)
- Kernighan and Ritchie-compatible C compiler
- shell that is functionally identical to the Bourne shell
- full multiprogramming (*fork+exec;* background jobs in shell: *cc file.c &*)
- full-screen editor vaguely inspired by emacs (modeless, autoinsert, and so on)
- more than 60 utilities (see list later)
- more than 100 library procedures (*atoi, fork, malloc, stdio, strcmp,* and so on)
- support for a hard disk but works quite well with just floppies
- programs to read and write MS-DOS diskettes
- full source code—about 54,000 lines—for operating system and all utilities (except C compiler, available separately)
- system can recompile itself (requires 640K and two floppies or one hard disk)
- kernel organization radically different from UNIX and much more modular
- software is not copy-protected

There are distributions on floppy for the PC/XT and the PC/AT and a generic distribution on mag tape. Here is a list of most of the utilities:

```
ar basename cat cc chmod chown
cmp comm cp date dd df dosread
echo grep gres head kill ln
login lpr ls make mkdir mkfs
mknod mount mv od passwd pr
pwd rev rm rmdir roff sh shar
size sleep sort split stty su
sum sync tail tar tee time
touch tr true umount uniq up-
date wc
```

Most notably absent is the *adb* debugger; the *ptrace* system call used for debugging processes is missing from MINIX. Also missing from the textbook is a description of the *ps* command; I don't know if it is in the distribution.

The MINIX software is not public domain, and you cannot distribute it commercially. But you can get it for your own use, run it on your system or systems, and experiment with variations in the operating system. The truly adventurous might consider porting it to other CPUs. You are not obliged to buy one copy of the source per CPU. Here is the text of the copyright notice from the source listing: "Copyright © 1987 by Prentice-Hall, Inc. Permission is hereby granted to private individuals and educational institutions to modify and redistribute the binary and source programs of this system to other private individuals and educational institutions for

FORTRAN PROGRAMMERS

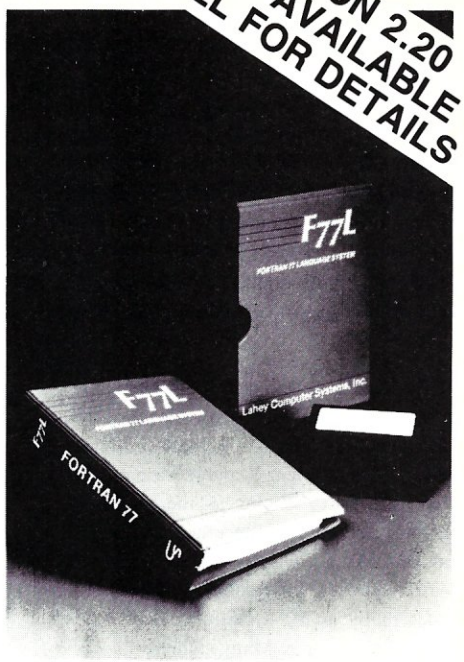
VERSION 2.20 NOW AVAILABLE CALL FOR DETAILS

Looking for the right PC FORTRAN language system? If you're serious about your FORTRAN programming then you should be using F77L - LAHEY FORTRAN.

"Lahey's F77L FORTRAN is the compiler of choice. It's definitely a 'Programmers FORTRAN,' with features to aid both the casual and the professional programmer. . . F77L compiled the five files in a total of 12 minutes, which was 4 times as fast as MS FORTRAN and an astounding 6 times as fast as Pro FORTRAN." - PC Magazine

Compare the features and performance of other PC FORTRANs with F77L and you will find that F77L is clearly the superior product.

- Full Fortran 77 Standard (F77L is not a subset)
- Popular Extensions for easy porting of mini and mainframe applications
- COMPLEX*16, LOGICAL*1 and INTEGER*2
- Recursion - allocates local variables on the stack
- IEEE - Standard Floating Point
- Long variable names - 31 characters
- IMPLICIT NONE
- Fast Compile - Increases productivity
- Source On Line Debugger (Advanced features without recompiling)
- Arrays and Commons greater than 64K
- Clear and Precise English Diagnostics
- Compatibility with Popular 3rd Party Software (i.e. Lattice C)
- Easy to use manual
- Technical Support from LCS
- *NEW FEATURE - NAMELIST*



F77L - THE PROGRAMMER'S FORTRAN

$477.00 U.S.

System Requirements: MS-DOS or PC-DOS, 256K, math coprocessor (8087/80287)

FOR MORE INFORMATION: (702) 831-2500

Lahey Computer Systems, Inc.
P.O.Box 6091
Incline Village, NV 89450
U.S.A.

International Dealers:
England: Grey Matter Ltd., Tel: (0364) 53499
Denmark: Ravenholm Computing, Tel: (02) 887249
Australia: Computer Transitions, Tel: (03) 537-2786
Japan: Microsoftware, Inc., Tel: (03) 813-8222

Editor's Choice
- PC Magazine

SERVING THE FORTRAN COMMUNITY SINCE 1967

MS-DOS & MS FORTRAN are trademarks of Microsoft Corporation.Pro FORTRAN refers to Professional FORTRAN a trademark of International Business Machines.

ISIS

CP/M MSDOS

COMPLETE SOURCE CODE INCLUDED!

ICXPDS: eXchanger now supports the 5 1/4"iPDS format. Manipulation of ISIS-II files using your computer system was never easier.

ICXMDS: Same as ICXPDS, but for MDS 8" systems.

IMXPDS: Reads/Writes 5" iPDS disks on PC's and AT's.

TELEDPLUS: Enhanced serial file transfer program for CP/M, ISIS, or MS-DOS.

ISE: Emulator gives the CP/M and MS-DOS user access to all the ISIS-II languages and utilities.

ACCELER 8/16: CP/M-80 emulator for MS-DOS. Enables PC's to run ISE. (no source code, V-20 incl.) **$89 each $250 any 3 above**

UDI: The 8086 ISIS Emulator runs all UDI applications. $300

ZAS Development Package: Z-8 and Z-8000 Assembler for CP/M, ISIS, and MS-DOS.

Request a catalog of our products!

Copyrights:CP/M Digital Research, Inc.
ISIS-II and iPDS Intel Corp. MSDOS Microsoft

Western Wares **303-327-4898**
Box C • Norwood, CO 81423

Rather Than Investing In Another Terminal INVEST IN "VT" Providing Superior TVI-925 Emulation On Your PC.

"Put Your PC's To Work On Your Multiuser."

Features:

- 4 VIDEO PAGES (MONO/COLOR)
- SUPPORTS DUAL HOST SESSIONS
- HOST STATUS LINE
- PROTECTED FIELDS
- PROGRAMMABLE CURSOR PAD
- 40 PROGRAMMABLE FUNCTION KEYS
- KEYBOARD MACROS
- PC/XT/AT COMPATIBLE

COST: COMPLETE **$79.95** POSTAGE PAID

DEMO AVAILABLE @ **$10**

DIGITAL DECISIONS SYSTEMS

229 Lakeview Ave. • Clifton, NJ 07011

201-772-6749

All Major Credit Cards Accepted

educational and research purposes. For corporate or commercial use, permission from Prentice-Hall is required. In general, such permission will be granted, subject to a few restrictions."—text, page 434.

Most UNIXs sold today are based on System V or 4.2BSD, so why did Tanenbaum choose to write a V7 clone? The reasons for choosing V7 had to do with the goals of the project, undertaken at the Vrije Universiteit in Amsterdam, Holland. In his own words the goals were:

> "1. To make a system that could be taught in operating systems courses. The key requirement was to make it small and relatively simple. System V and 4.3 are much, much larger.... As far as the professor teaching the course is concerned, additional tens of thousands of lines of code to get job control, etc., is a bad tradeoff. 2. To provide an alternative to MS-DOS for students who have a PC at home. Many students have a PC at home, usually without a hard disk. I wanted something that would work well without a hard disk. MINIX works fine without a hard disk, but I can hardly imagine System 5 or 4.3 running without a hard disk."

How do you get a copy of the software? Buy the book from your favorite bookstore or computer shop. In the book you will find an order form for the software. The book lists for $35; the software is $79.95 per set, including the source code for everything except the compiler.

The lower levels of MINIX are a message-passing kernel, which is slower than a real UNIX implementation (such as XENIX or PC-IX) might be. Alas, I didn't get to run any benchmarks at the show. Tanenbaum asserts (page 68) that message passing is inherently slow compared to other methods of intrakernel communication. But slow or not, MINIX has the inestimable advantage of source code availability.

I mentioned that the brave might consider porting MINIX, and Tanenbaum drops a few hints throughout the book, though he does not offer a porting guide. Be aware, though, that MINIX knows a *lot* about the IBM PC architecture. Much of the machine-dependent stuff (but not 100 percent) is identified by *#ifdef i8088* or by file names such as **88.s*. Additionally, MINIX neither swaps nor pages processes. It lacks a debugger, and it seems to lack the *ps* command. There is no *mail* and no *uucp*. In short, it was written as a demonstration system and as an essentially single-user OS for the IBM PC; porting it to make a production operating system on another architecture is certainly possible but emphatically nontrivial.

As I said, I haven't seen the full sources yet, though I have seen the system running. I should have my tape in hand by the time you read these words and will try to follow up with more information in a future column. I think this package has the potential to get many technically oriented PC owners (including *Micro/Systems Journal* readers), tinkerers, or (dare I use the word?) PC *hackers* running a full-function operating system at hobbyist prices. I welcome MINIX as a significant event in the UNIX and PC worlds.

ROLL YOUR OWN AT&T 3B COMPUTER FROM SBC?

The AT&T 3B line of computers has achieved wide acceptance among business users of UNIX. Now you can build your own 3B and have it running on the fast VMEbus (mentioned in this column in the January/February 1987 issue.) And you can use boards from many different manufacturers.

AT&T Technology Systems is selling a single-board computer using the same CPU, memory manager, and math chip as is featured in the 3B2 and larger models of the 3B family. The AT&T WE 321SB single-board computer features a 32100 CPU running at 14 MHz (soon at 18 MHz), a WE32101 memory-management chip, and a WE32106 floating-point chip called the Math Acceleration Unit (MAU). The board also features a full megabyte of on-board dynamic 4RAM with parity, a quarter megabyte of EPROM with a bootloader and diagnostic PROM set, two serial ports, and three programmable counter-timers. That's the hardware.

No SBC is much use without software. It would be almost impossible for people in the field to port UNIX to a new product such as this. So AT&T provides a port of AT&T System V UNIX custom-made for this board. There are drivers for several popular VMEbus controllers and a manual on writing your own device drivers. You will need a Signetics/Phillips PG3103 disk controller to boot the system initially; this board runs both 5¼-inch floppies and SA450/ST506 small Winchester disks. There is also support for DUAL Systems' fast VMEbus SMD disk controller and the Ironics IV1624 eight-line serial board, with other drivers "in the works."

There is a short note on what you will need to build a complete system (at least a WE 321SB, a system controller, a disk controller and hard disk drive(s), card cage and backplane, power supply, cabinetry, and possibly some more memory). The board is also supported by AT&T training courses on driver writing (in preparation) and about 20 AT&T manuals. The software (UNIX system, device drivers) can be had either in source or binary form; the binary is much cheaper. You could buy the UNIX system in binary and the device drivers in source form if you wanted to modify the drivers or adapt them to new devices.

Once you get your system working, of course, you will have earned two benefits. One is the satisfaction of having built a fast, multiuser UNIX system with boards from several different vendors and with the ability to use boards from almost all other VMEbus board builders. This is what attracted many people to the S-100 bus and will probably interest many readers of *Micro/Systems Journal*. The second benefit is the variety of software; the wide range of commercial UNIX software for the 3B2 computer system will run on your home-built, and most of the wide range of public-domain software for UNIX systems in general will also be available.

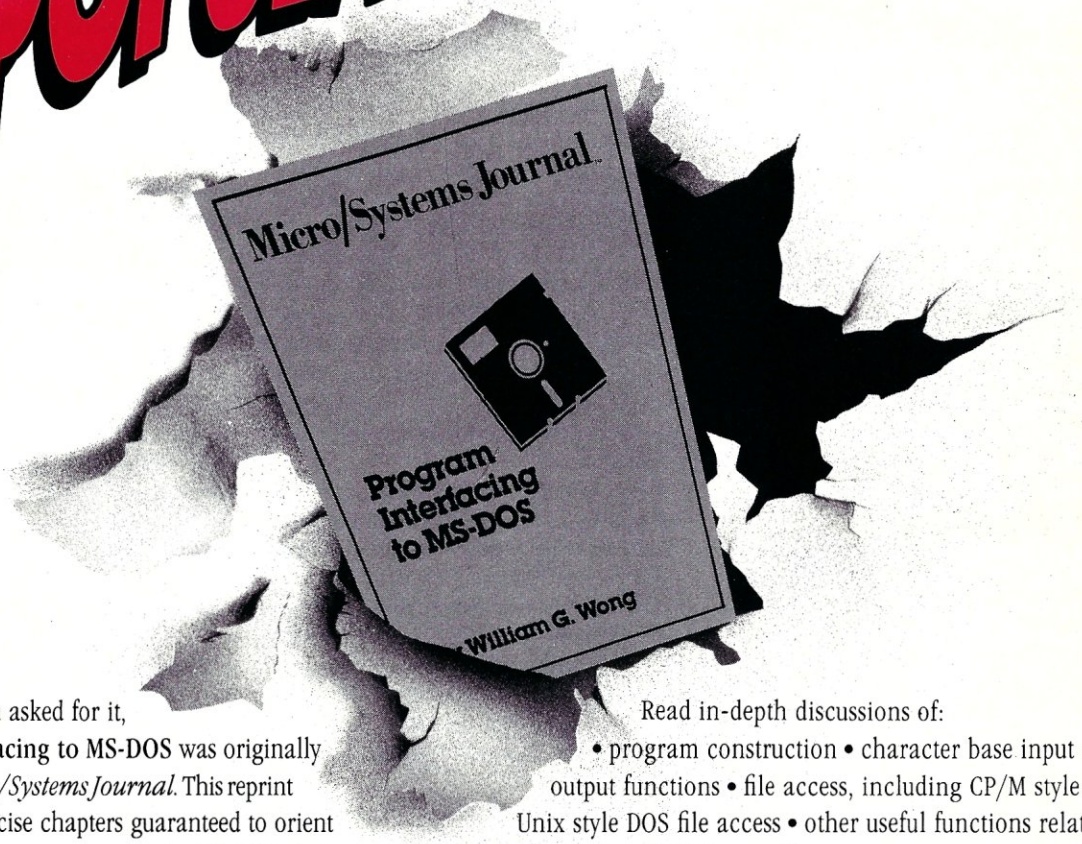
But all this stuff is industrial grade, and it's not cheap. AT&T does not like to discuss pricing over the phone, or so it tells me, so if you want the prices, I'm afraid you'll have to contact your local AT&T rep. Prices are not cheap but are comparable to other vendors' VMEbus products. If you aren't up to speed on UNIX System V device drivers, you would certainly need working examples from which to start any new driver development; consider getting the source for the existing drivers or at least demo versions.

To get more technical details on building your own VMEbus-based 3B computer, contact AT&T Technologies at (800) 372-2447. In Canada, call collect at (215) 266-2973. The mailing address for this product is AT&T, Dept KB, 555 Union Blvd., Allentown, PA 18103.

That's all for this issue. I welcome letters and electronic mail on these and other topics, including suggestions for future columns. §

Ian Darwin is director of Research and Development for SoftQuad Inc., a company providing supported troff publishing software for UNIX. He and his wife live in a rural setting, north of Toronto.

BACK BY POPULAR DEMAND



Because you asked for it, **Program Interfacing to MS-DOS** was originally featured in *Micro/Systems Journal*. This reprint provides ten concise chapters guaranteed to orient any experienced programmer to the MS-DOS environment.

These articles reprinted together for the first time:

- The Program Segment Prefix, Access to DOS, and How to Terminate a Program
- Dealing with Character Input and Output Functions
- Basic File Access Functions
- Basic File Access
- Programming MS-DOS
- Device Drivers—Why and How
- Device Drivers, Structures
- A Memory Disk Device Driver and a Printer Device Driver (both provided on disk with full source code)
- Why Move to DOS 3.X?

Read in-depth discussions of: • program construction • character base input and output functions • file access, including CP/M style vs. Unix style DOS file access • other useful functions related to memory allocation and program execution.

Program Interfacing to MS-DOS also contains sample program files and a detailed description of how to build device drivers.

The complete ten-part **Program Interfacing to MS-DOS** reprint is now available for only $29.95. It includes the device driver for a memory disk and a printer device driver, both on disk with macro assembly source code.

To order, return this coupon to: M&T Publishing, Inc., 501 Galveston Drive, Redwood City, CA 94063. **Or, CALL TOLL-FREE!**

Yes! Please send me **Program Interfacing to MS-DOS**

for $29.95 _____

Tax (CA only) _____

Shipping, add $2.25 per item _____

TOTAL _____

☐ Check Enclosed. Make payable to M&T Publishing.

Charge my: ☐ VISA ☐ Master Card ☐ Am Ex

Card # _____ Exp. Date _____

Name _____

Address _____

City _____ State _____ Zip _____

33A

800-533-4372 (IN CA 800-356-2002) ASK FOR PROGRAM INTERFACING TO MS-DOS, ITEM #166.

The Public Domain Software Forum

Charles H. Strom

What Is Public Domain?

Hmm, only on my second column and already departing from my self-imposed structure. Although I promised to tailor the column to the issue's theme, I feel the importance of this topic warrants a diversion. I hope you feel likewise.

There is a long-standing confusion over some terms used by authors of software making the rounds of the remote systems, telecommunications services, and nonprofit software libraries. Although I am not an attorney, I have investigated the situation and will attempt to clarify a few terms, hopefully preventing some future catastrophes. Here is an excerpt from a currently available program:

"This program has been placed in the Public Domain by the author and copies may be freely made for non-commercial, demonstration, or evaluation purposes. Use of these subroutines in a program for sale or for commercial purposes in a place of business requires a $20 fee be paid to the author at the address above. . . ."

The author of this program has released it into the public domain and then applies restrictions on its use. Therein lies the conflict. The term *public domain* implies just that—the program belongs to everyone. Readers would think they had the perfect right to copy a public-domain program verbatim for any use desired, such as advertise and sell it in *Micro/Systems Journal*. True, in the above case the author has applied specific restrictions that could support his case of ownership, but it is misleading to have used those magic words *public domain*. It is important to copyright a program under the applicable statutes as well, unless you are willing to abandon your rights to the program. An unscrupulous soul may argue that a mere common-law copyright was abandoned by the owner's actions in generally disseminating the program to the public. I am sure that, because of this legalistic masterpiece, we need no longer worry about ambiguities as in the above example (hah!) and I can turn my attention to the real purpose of this column—to talk about software!

TOP TEN #2—CED

I'd like to sing the praises of CED, a DOS command editor written by Chris Dunford. CED has several functions, including expanded command-line editing (at the DOS A> prompt), recall of previously entered commands, command "synonyms," and so on. CED installs in IBM PCs and compatibles under Version 2.0 or later of MS-DOS. It installs as a memory-resident (terminate-and-stay-resident, or TSR) program and uses about 16K, mostly devoted to buffers. Let's look at CED's functions.

Once installed, CED supplies a much-enhanced command-line edit facility. While entering a command at the DOS prompt, you can use the arrow keys to move the cursor anywhere along the line and overtype, delete, or (using the Insert key as a toggle) insert characters anywhere along the line. This is like having a mini-text editor right in DOS. To clarify, let's say I have keyed in the following:

```
C)\ WORD \WORK\DOCS\ARTICLE.JUL
```

and discover to my chagrin that I made a typo and my subdirectory is not *\WORD\DOCS* but rather *\WORD\DOC*. Before CED, I had to use the destructive backspace key to get to that point and rekey the correct directory name. Now all I need do is use the left arrow key to locate the cursor over the offending *S*, and press the Delete key, correcting the entry. You can imagine the time-savings this CED function affords.

CED provides a buffer that stores a history of the last *n* commands, where *n* is related to the length of commands and the size of the buffer (which can be modified). This buffer is circular. In other words, use of the up and down arrows allows scrolling through the last *n* commands quickly and easily. At any point, you can press a Return and execute the command, use CED's editing feature to modify it before execution, and so on. This is a big time-saver.

The last feature I would like to discuss is the synonym capability. Taking an example directly from the CED documentation, suppose you type the following when invoking your communications program:

```
CD \TALK
TALK CIS /C63
CD \
```

CED will allow the following:

```
CED SYN CIS CD \TALK^
    TALK CIS /C63^CD \
```

Note that the ^ character is the command separator. Now when you type in "CIS," CED will enter everything to the separator and issue a carriage return. It will follow with the next command when input is requested and so on. Several related commands are used to enhance the synonym capability, too.

Chris Dunford has done a super job with this program, and I would not be without it. It has several additional features that I will not cover here, but one of the nicest things about CED is that it is in the public domain, so you can obtain this program from CIS, GEnie, private remote systems, or on PC/Blue diskette and try it out, read the documentation, and explore its complexities to your heart's content with no strings attached whatsoever. Chris also offers a low-cost, commercial version called PCED. You can reach him at 10057-2 Windstream Dr., Columbia, MD 21044.

REFEREE

I have been using this DOS memory-resident program (also known as a TSR) for several months. Referee is distributed by Persoft, 465 Science Dr., Madison, WI 53711. The list price is $69.95, though it is available through retail channels for considerably less. True, Referee is not a shareware program, but it is low cost and offers significant advantages to MS-DOS users. Because you cannot try before buy-

ing, I will summarize Referee's features.

Referee is basically a TSR that manages other TSRs. Its resident portion (REFWATCH) is loaded via AUTOEXEC.BAT before any other TSRs. Unfortunately, the program cannot supervise drivers that load via CONFIG.SYS. It is a simple matter to invoke Referee at any time thereafter to take a snapshot of the TSR situation at that moment. Once you invoke Referee, you have several options, including activate/deactivate RAM-resident programs, unload programs (thereby freeing memory), and so on. One of the bugaboos of TSRs is that they are RAM hogs. Imagine loading MODE.COM and a laser-printer driver to do a printing job and then loading a spreadsheet program. Before Referee, I had to reboot the computer to free up the space. Now all I need to do is invoke Referee and unload the drivers. The time-savings and added convenience are significant.

Another feature is called Ramteams. This is an automatic facility that allows users to specify which of the many TSRs at their disposal should be activated or deactivated to run a particular application. Why would you want this capability? How about selectively enabling or disabling a memory-resident spelling checker such as Turbo Lightning, for example? You certainly do not want it active when doing a sorted directory listing. Another obvious application is utilizing several printer drivers depending on the specific software you are running.

Last, there is a memory-resident-limited version of Referee (remember that although you need a resident kernel called REFWATCH, Referee itself is invoked as a regular program). Sideline Referee can be invoked as a pop-up within an application to activate or deactivate TSRs.

Several shareware and public-domain MS-DOS utilities are available that perform some of the functions of Referee (TSRCOM, for example), but none have the versatility and ease of use required for such a program. Considering its reasonable price, Referee is a worthwhile addition for those who are tired of juggling their memory-resident programs and are faced with memory-space limitations. §

Charles Strom's first exposure to computers was in high school, running a then-state-of-the-art IBM 1620 with 20K of core and read/punch cards. He built an IMSAI 8080-based microcomputer in 1977 and was bitten by the micro bug. Although a chemist by training, Charles is a computerist by choice. Current interests include scientific/technical word processing, communications software, and desktop publishing. Charles is the author of numerous review articles and a sysop on the GEnie national time-sharing systems.

PC-SIG CD ROM

PC-SIG, the largest commercial distributor of PC public-domain software has begun shipping its CD ROM. The package consists of a Hitachi CD ROM player, interface card, and CD ROM disk containing the equivalent of 605 floppy disks of MS-DOS software (550 MB). A standard floppy disk is also provided with a driver program for the player.

Installation is simple and straightforward. The disk is organized in subdirectories for each volume in the PC-SIG library. It typically takes about 2 seconds to find a directory, and once in the directory, access time is equivalent to a hard disk. The software can be copied from the CD ROM disk to any other medium. The software driver provided by PC-SIG does not have any searching facilities. No doubt some enhanced drivers for the system will soon appear in the public domain.

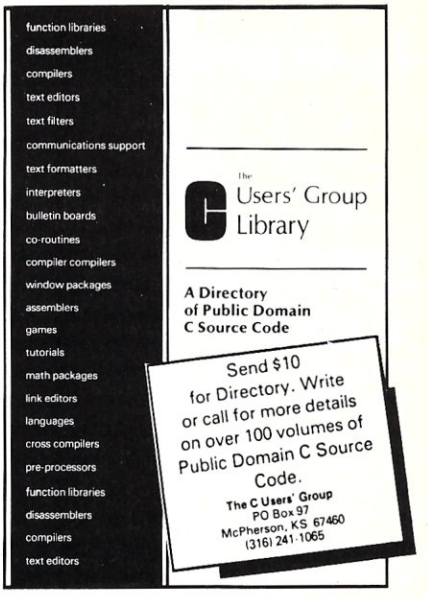
The cost is $1,500 for the hardware and $300 for the CD ROM disk. Considering that the equivalent in disks from PC-SIG (at $6/disk) is $3,650, this represents a significant savings for anyone or any organization wanting the entire library. The CD ROM approach means that the entire library is on-line and you do not have to search through 605 floppy disks.

SIG/M AND PC/BLUE REPORT

Steve Leon has retired as the editor/compiler of disks for SIG/M. Steve had put together more than 100 of the SIG/M disks and had done a masterful job. I am sure all users of CP/M systems are in his debt for the work he did, all on a voluntary basis. The SIG/M group is searching for a volunteer replacement to fill the void. Until this happens no new SIG/M disks will be released.

Hank Kee, the editor/compiler of PC/Blue disks has been busy during the last few months and hence no new disks have been released since the report in the last issue of *M/SJ*. A new batch of disks should be released shortly.

—Editor



NEW

FOR YOUR BIGBOARD

ONE MEGABYTE RAM DISK ON THE STD BUS
Includes: STD adapter pcb & connector, 1 MB RAM pcb & connector and software. Price: $125.00.
INTEGRATED BIOS, reads and writes any 5" and/or 8" format.

INCLUDES:

- CONFIGuration program that lets you install any new floppy disk format INTERACTIVELY.
- FORMATTER allows you to format almost any diskformat.
- PC-COPY reads and writes PC diskettes on your Bigboard II.
- MONITOR EPROM with serial keyboard and translate table.
- 300 page ZCPR2 manual.
- 60 page Bigboard II tech manual.
- BOOTABLE DISK contains free ZCPR2 and P2DOS system. With TIME and DATE stamping. BIOS also has provisions for 256K RAM disk, Centronics, System in EPROM AND 1 MB RAM DISK.
- WINCHESTER FORMATTER and SYSGEN. Supports XEBEC, W-D and Adaptec type controller. Subdivides into any specified number of drives. Price: $99.95 (specify disk).

TAKE BOTH 1 MB AND BIOS FOR $199.95

ANDY BAKKERS

de Gervelink 12 ● 7591 DT Denekamp
The Netherlands ● Tel: 31-5413-2488
FIDO Net 500 Node 100. Please pay with US$ Money Order. MC or VISA welcome.

LANScape

B.J. Hall & Michael Cherry

Last issue's LANScape provided a foundation for future discussions on all the complex factors involved with networks and networking. This month's column focuses on the cable plant that connects the various devices on the local area network (LAN).

As the medium on which network communications occurs, the cable plant is one of the key elements of the LAN. Despite the importance of the cable plant to the success of the LAN, it is one of the most overlooked elements. As stated last month, the four factors that determine a networks superiority are:

- reliability
- ease of maintenance
- ease of expansion
- performance

These four factors apply to cable, just as they apply to the LAN hardware and software. Let's look at the different types of cable used for LANs.

UNSHIELDED TWISTED PAIR

Unshielded twisted pair cable is commonly referred to as telephone cable. Two wires, wrapped individually in an insulation, are twisted together to form a pair of wires. This cable is called unshielded as there is no protective barrier around the wires to prevent the radiation of signals to or from the twisted pairs.

Miles of twisted pair cable exist in office buildings today carrying telephone conversations, but for the transmission of data, unshielded twisted pair is not very reliable. As there is no shielding, the cable can pick up interference from other cables, including nearby power cables.

For voice communications, unshielded twisted pair cable is easy to maintain. It can be easily spliced and repaired. For LANs, maintenance may not be as easy because most LANs do not perform well on twisted pairs with too many splices.

Unshielded twisted pair is easy to expand. It is easy to pull through a building because it is not thick, and therefore it is easy to install. In most cases, there may even be spare cables already routed through most of the building.

Finally, consider performance. The IBM Token-Ring LAN for example, can operate at 4 megabits (four million bits) per second on unshielded twisted pair. On shielded twisted pairs, the IBM Token-Ring will in the future operate at 16 megabits per second.

The advantage of unshielded twisted pair cable is that it is inexpensive, and in many cases, it is readily available in the building. Its disadvantage is that it does not offer good reliability or performance on high-speed nets.

SHIELDED TWISTED PAIR

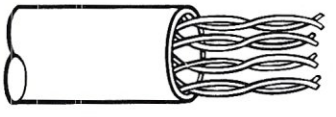
Shielded twisted pair cable is similar to unshielded twisted pair, except that braided wire shield is wrapped around the twisted pairs. This shielding increases the reliability and performance of the twisted pairs. The very element that adds this increased reliability and performance, however, impacts the ability to install and maintain the cable—it is difficult to splice this cable and maintain the integrity of the shielding, without actually using connectors.

The advantage of shielded twisted pair cable is that it is reliable and offers excellent performance. The disadvantage is that it is expensive and can be difficult to install.

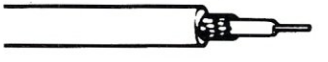
COAXIAL CABLE

Coaxial cable consists of a solid inner conductor surrounded by an outer braided shield. The two conductors are separated by an insulating layer.

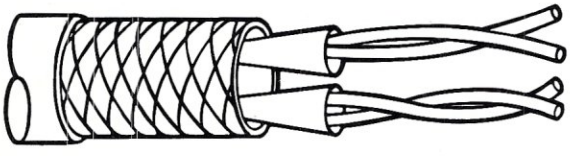
As with shielded twisted pairs, the shielding makes coaxial cable very reliable. It is immune to interference from other signals and electrical noise. In addition, the wide bandwidth of the coaxial cable permit a large number of signals to be carried on a single cable—for example, co-



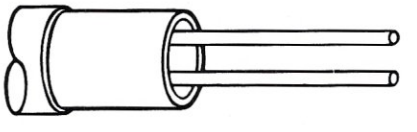
Unshielded Multiple Twisted Pair



Coaxial Cable



Shielded Multiple Twisted Pair



Fiber Optic Cable

axial cable is capable of carrying the large number of television signals that cable TV operators send throughout their system. This means that the performance of coaxial cable is very good.

Again, maintenance and expandability may be an issue. When splicing coaxial cable, it necessary to use connectors to ensure that the shielding continues to perform its function. It can also be difficult to install.

The advantage of coaxial cable is that it is well shielded, ensuring performance and reliability as well as a wide bandwidth. The disadvantage of coaxial cable is that it can be difficult to install.

FIBER OPTIC CABLE

In addition to these copper cables, there is fiber optic cable. Fiber optic cable is actually a thin glass cable, jacketed to protect it from breakage.

Fiber optic cable provides a highly reliable cable that offers high performance and bandwidth, but it is expensive, and despite recent advances, can still be difficult to install and splice. This cable is usually too expensive for use on a LAN except for linking physically separate LANs.

CHOOSING A CABLE

In Table 1, you will note that in some cases you have a choice as to which cable to use. In actual practice, however, each LAN has a cable preference that is recommended to ensure performance and reliability.

In designing a LAN, you are not always given the best of possible conditions—for example, a company may already have a sizable investment in an existing cable plant but may want to utilize a LAN that is not compatible with that cable type. As a specific example, we encountered a company that moved to buildings that were prewired with unshielded twisted pair cable for both voice and data. This company already owned an ARCNET LAN, which typically uses coaxial cable. In this case, devices called baluns were available to permit the connection of the ARCNET hardware on twisted pair cable. These baluns convert from a BNC (coaxial) connector to an RJ-11 (twisted pair) connector. In addition to allowing the different connectors to mate, the balun also matches the impedance of the cable to permit the twisted pair cable appear to the ARCNET hardware as coaxial cable.

Although devices such as baluns are available, it is necessary to keep in mind that the manufacturer of a particular LAN has many reasons for selecting a cable type, and whenever a deviation is made from these standards, performance and reliability may be negatively affected.

Table 1.

LAN	Unshielded Twisted Pair	Shielded Twisted Pair	Coaxial
AT&T STARLAN			
ARCNET	1	1	2
Ethernet			3
IBM PC NETWORK			4
IBM Token-Ring (802.5)	5		
PROTEON PRONET - 10			
PROTEON PRONET (802.5)	5		

Notes
1. Requires Baluns to match coaxial impedence **2.** Type RG-62/U coaxial cable **3.** Type RG-58/U coaxial cable **4.** Type RG-59/U and RG-6/U coaxial cable **5.** Severe distance and device restrictions

= Yes = No

Table 2.

LAN Success Factor	Unshielded Twisted Pair	Shielded Twisted Pair	Coaxial
Performance			
Expandability			
Maintainability			
Reliability			

= Excellent = Good = Poor

Another consideration with cable is the type of insulating jacket that surrounds the cable. Polyvinyl chloride (PVC) insulation gives off poisonous gases in a fire and therefore can only be installed in conduit. PVC cable is usually inexpensive, but the cost of installing conduit may exceed the savings in cable cost. On the other hand, Teflon-jacketed cable does not have to be installed in conduit. Teflon insulation can be installed in plenum ceilings (false or drop ceilings) and still stay within the fire codes. Although teflon cable is more expensive than PVC cable it may be cheaper to install because you do not need conduit.

Regardless of what type of cable jacket you choose, you must ensure that it is installed in a manner that meets the relevant fire and building codes in your area.

In conclusion, no one cable type is necessarily right or wrong for your LAN. The IBM Token-Ring LAN, for example, can work on unshielded twisted pair cable (referred to by IBM as Type 3 cable) or a shielded twisted pair cable (referred to by IBM as Type 1 cable). The difference in use is the configuration rules that apply. These rules may affect the:

- number of devices attached to the LAN
- number of access points that the devices connect through
- distances between the devices on the LAN
- total length of the cable plant

It is therefore important that you keep the success factors of a LAN in mind and select the cable that offers high reliability and high performance. Unfortunately, this may require that you make trade-offs. Once the choice has been made, you must design a wiring plan that permits both ease of maintenance and further expansion. This is not a easy task, and as mentioned in the beginning of this article, is often overlooked. As a rule of thumb, the design and installation of the cable plant can cost up to 40 percent of the cost of all the other LAN components. In one case, the installation of cable for a 13-workstation LAN cost $18,000. Although this particular case is extreme, it illustrates the expense that can be associated with installing cable correctly and being in compliance with building and union codes.

Next Month: LAN Topologies. §

B.J. Hall and Michael Cherry operate HallComm Network Services, a company devoted exclusively to designing and implementing LAN systems. HNS is located at 8101 E. Prentice Ave., Ste. 304, Englewood, CO 80111; (303) 770-6387.

New Products

When contacting vendors, please mention that you read about their products in Micro/Systems Journal.

Manufacturers who would like to have their hardware products listed here should send their news releases to The Editor, Micro/Systems Journal, *P.O. Box 1192, Mountainside, NJ 07092.*

PC-COMPATIBLE HARDWARE

MULTIUSER/MULTITASKER FOR 386-BASED SYSTEMS

Xmark Corp. has introduced PCplex (tmk), a plug-in board for systems using the 80386 processor that provides multiuser, multitasking functions. It supports up to three users running under DOS 3.1. The board fits a half or full slot and utilizes low-cost ASCII terminals connected to existing ports. The operating system is ROM-resident and allows standard PC-DOS software to run in a multiuser/multitasking mode without changing the software, files, or directories.

The price is $199, and a version that adds two additional COM ports costs $399. For information contact Xmark Corp., 3176 Pullman St., #119, Costa Mesa, CA 92626; (714) 556-9210.

DATA ACQUISITION PROCESSOR FOR LOTUS 1-2-3

Microstar Laboratories has released a plug-in data acquisition board and software that allows direct, real-time interaction between Lotus 1-2-3 and the board. It can be controlled directly from 1-2-3, eliminating the need for programming. The board contains an 80186 processor and a ROM-resident multitasking operating system to perform complex real-time computations such as Fourier transforms, digital filtering, and thermocouple compensation, feeding the processed data to 1-2-3.

The processor can also operate in the background while 1-2-3 does computations or graphing. The data acquisition processor samples at speeds up to 17,500/sec with data buffered on the board. The price ranges from $1,595 to $2,500 depending on hardware configuration and software provided. For information contact Microstar Laboratories, 2863 152nd Ave. NE, Redmond, WA 98052; (206) 881-4286.

BUS DIAGNOSTIC MONITOR FOR PC, PC/XT, or PC/AT

The Crowcard I-Bus Monitor, from Applied Physics, allows hardware analysis using the data available on the system bus of a PC, PC/XT, or PC/AT. The bus' activity can be observed without interfering with the normal operation of the system. The monitor consists of a plug-in board and software. The board contains 50 LEDs, grouped by function, to monitor the lines on the bus.

The price is $249. For information contact Applied Physics Inc., 1291E Cumberland Ave., West Lafayette, IN 47906; (317) 497-1718.

LINKER FOR PCS TO COMMON PRINTER

EasyPRINT (tmk) is a software package with cables that allows up to 14 PC-compatible satellite machines connected to a central hub computer to share common printers using RS-232 serial ports. It supports simultaneous printing to as many as four printers or plotters attached to the serial or parallel ports of the hub computer. It is transparent to users at both the satellite and hub systems, and there are no switches to set.

A two-PC starter kit costs $159.95 and includes software, manuals, and 30-foot cable. A three-PC kit costs $349.95, and a seven-PC kit costs $899.95. A two-port board costs $219.95, and a six-port board costs $489.95. For information contact Server Technology, 1095 E. Duane Ave., #103, Sunnyvale, CA 94086; (800) 835-1515 or (408) 738-8377.

OTHER HARDWARE PRODUCTS

TRACKBALL

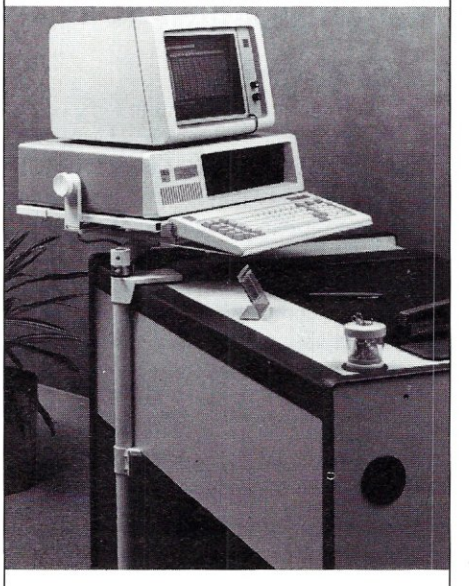
The Fulcrum Trackball Plus is an enhanced trackball device that includes six user switches and is designed for use with application software such as AutoCad. It has a unique function, allowing swtiching between two cursor positions actuated by one of the user switches; the operation is transparent to the software. This is useful for switching between the menu and CAD position.

Ten emulations are supported, including the popular mouse and digitizer formats. The user-switch configuration can be set up without utilities. All switches feature tactile feedback.

The unit retails for $95 and is available from Fulcrum Computer Products, 451 Allan Ct., Healdsburg, CA 95448; (707) 433-0202.

SPACE-SAVING COMPUTER WORK STATION

The Flexarm is a computer work station that removes the computer from the desktop, swinging it out of the way when not in use. It also allows the computer to be used by more than one person. It supports up to



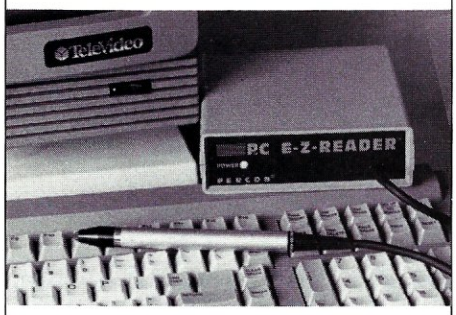
150 lbs; can turn 360 degrees; and is adjustable in height, position, and viewing angle. The keyboard retracts when not in use. For information contact Screen Data Corp., 80 S. Jefferson Rd., Whippany, NJ 07981; (800) 248-1212.

BAR-CODE READER FOR TELEVIDEO TERMINAL

The Percon E-Z Reader is a plug-in device that installs between the keyboard and terminal of a TeleVideo terminal. It requires no extra power. Data from the unit appears to the computer as if it had been

entered from the keyboard. The keyboard can still be used.

The unit decodes all popular bar codes, including Code 39, Codabar, Uniform Product Code A and E, and Interleaved 2

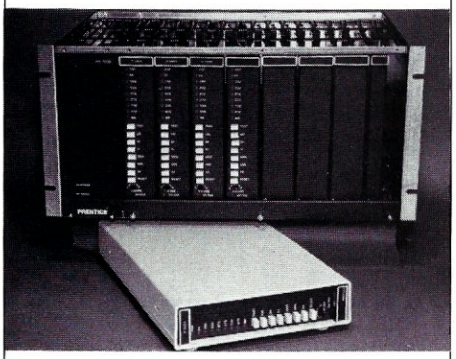


of 5. The unit consists of a decoder box and metal wand. It has automatic self-test upon power-up and produces a beep as it reads a label.

The unit has a 30-day trial period and comes with 2-year warranty. For more information contact Percon, 2190 W. 11th Ave., Eugene, OR 97402; (503) 344-1189.

EXPANDABLE INTELLIGENT 2,400-BPS MODEM

The Prentice P-2424 2,400-bps modem offers full-duplex asynchronous and synchronous operation; provides Hayes AT command set compatibility; and meets Bell 103, 212A, and CCITT V.22bis specifications. It accepts multifunction add-on boards and includes a voice/data switch allowing switching between data and voice during the same phone call. Microcom's



MNP Class 3 option can be added. Stand-alone and rack-mount units are available. The unit includes diagnostic indicators and several test modes. Stand-alone and rack-mount units can be interchanged and intermixed with other Prentice modems in the same rack enclosure.

For information contact Prentice Corp., 266 Caspian Dr., Sunnyvale, CA 94088; (408) 734-9810.

UNINTERRUPTIBLE POWER SUPPLIES

Clary has introduced the ONGUARD line of uninterruptible power supplies with a very small footprint (9-inches wide × 18-inches deep), in 400VA, 500VA, and 600VA sizes. They provide sine-wave output, are designed for continuous use, and are claimed to be virtually unaffected by nonlinear loads such as switching power supplies.

These UPS's include fast transient response and a computer-compatible interface for alarms and remote on/off control. They are designed to meet IEEE-587, FCC, and U.L. requirements. For additional information contact Clary Corp., 320 W. Clary Ave., San Gabriel, CA 91776; (818) 287-6111.

3270 16-PORT CONTROL UNIT

Local Data has announced Datalynx/3174, a 16-port cluster controller connecting asynchronous networks, PCs, terminals, and printers to IBM 3270 mainframes. It emulates IBM's SNA 3174, 3274, and 3276 control units. It is modular, so a user can start with only four ports. The unit and accompanying software provide many features in addition to IBM compatibility.

A 4-port unit costs $3,000, and a 16-port unit costs $6,000. A 16-port unit that supports remote IBM type A coax terminals and provides X.25 PAD interfaces and extended network management capabilities costs $6,500. A protocol converter to support IBM 3178/3179 terminals costs $695-$795 (depending on quantity), and the software costs $195. For information contact Local Data Inc., 2772 Toledo St., Torrance, CA 90503; (213) 320-7126.

ON COMMAND:

Writing a Unix-Like Shell for MS-DOS by Allen Holub

Learn how to write shells applicable to MS-DOS, as well as to most other programming environments! This book and disk include a full description of a Unix-like shell, complete C source code, a thorough discussion of low-level DOS interfacing and significant examples of C programming at the system level.

Supported features include: • Read • Aliases • History • DOS compatible support • C-Shell-based shell scripts

The Unix-like control flow includes: if/then/else; while; foreach;switch/case; break; continue.

For IBM PC compatibles. All source code is included on disk.

/UTIL

When used with the shell described in *On Command,* these utility programs and subroutines provide a fully functional Unix-like environment! Utilities include: cat; cp; date; du; echo; grep; ls; mkdir; mv; p; pause; printevn; rm; rmdir; sub; and chmod. Complete source code and manual are included.

Receive *On Command* together with */UTIL* for only $59.95!

On Command	**Item #163 $39.95**
/UTIL	**Item #161 $29.95**
On Command with /UTIL	**Item #164 $59.95**

TO ORDER: Return this coupon with your payment to: M&T Books, 501 Glaveston Dr., Redwood City, CA 94063. Or, call **TOLL-FREE** 800-533-4372 Mon-Fri 8a.m.-5p.m. In CA call 800-356-2002

YES! Please send me **On Command: writing a Unix-Like Shell for MS-DOS** with disk for $39.95 _____

Send me **/UTIL** for $29.95 _____

Send me both **On Command** and **/UTIL** for only $59.95 _____

Subtotal _____

CA residents add sales tax _____ % _____

Add $2.25 per item for shipping _____

TOTAL _____

Name _____

Address _____

City _____

State _____ Zip _____

☐ Check Enclosed. Make payable to M&T Publishing.
Please Charge my ☐ VISA ☐ M/C ☐ AMEX

Card No. _____

Exp. Date _____

Signature _____

33D

Micro/Systems Journal®

Back Issues

The following back issues are available at $5.00 for 1 issue, $4.50 each for 2-5 issues and $4.00 each for 6 or more issues.

MARCH/APRIL 1985 (Vol. 1, No. 1): Bringing up CP/M Plus; Assembly Language Extensions for MS-Basic; New Tricks for CP/M2.2; Building an IBM/PC or XT Clone; Extended Single Density Storage; Variable Size Arrays in C; REVIEWS: dBASE III and 16-Bit Lisp & ProLog—Part I.

MAY/JUNE 1985 (Vol. 1, No. 2): Build an S-100 to PC-Bus Converter; Interfacing to MS-DOS—Part I; Loadable Drivers for CP/M2.2; Roll Your Own PC-Clone; Bringing up ZCPR-3; C & Godbout Disk-1 Controller; Writing Translation Programs in C and Turbo Pascal; REVIEWS: 16-Bit Lisp & Prolog—Part II.

JULY/AUGUST 1985 (Vol. 1, No. 3): Structured Programming With Microsoft M80 Assembler; Local Variables in Forth; Interfacing to MS-DOS—Part II; Data Translation with Turbo Pascal; Implementing Sets with Bit Operations in C; A Unix Mail List System; REVIEWS: Scientific & Technical Word Processors—Part I; Macrotech MI-286 S-100 CPU Card; Slicer System; Concurrent PC-DOS; Coherent Operating System.

SEPTEMBER/OCTOBER 1985 (Vol. 1, No. 4): Who Prints on Printer with CCP/M; Interrupt Borrowing with Turbo Pascal; Upgrading CompuPro I/O Boards; Program Interfacing to MS-DOS—Part III; SIG/M & PC/Blue Public Domain Software Distribution Points; TurboDos INSTALL program; REVIEWS: CompuPro 10+; TEX.

NOVEMBER/DECEMBER 1985 (Vol. 1, No. 5): TurboCharge Your 8086/8088 Computer; Faster Floating Point Math with C; Bringing up CP/M-86K; Program Interfacing To MS-DOS—Part IV; Advanced Machine Level Interface Techniques For Turbo Pascal; Context-Independent Macros for C; Build Your Own PC Program Library Using Public Domain Software; dBASE-II Speed Techniques—Part I; Tuning Unix Program; REVIEWS: Peak 68K8-CP; Eureka For CP/M-80; Scientific Workstations.

JANUARY/FEBRUARY 1986 (Vol. 2, No. 1): Implement PC-DOS on Non-IBM Compatible Computers—Part I; TurboCharge Your 8086/8088 Computer—Part II; Assembling An AT Clone; Build An S-100 EPROM Emulator; Transferring Files Between CP/M and MS-DOS Systems; Program Interfacing to MS-DOS—Part V; dBASE-II Speed Techniques; Unix Public Domain Software; REVIEWS: Scientific Word Processors—Part II.

MARCH/APRIL 1986 (Vol. 2, No. 2): Implementing PC-DOS On Non-IBM Compatible System—Part II (Conclusion); Program Interfacing To MS-DOS—Part VI; Rolling Your Own PC/XT/AT Clone; Converting From CP/M To MS-DOS On The PC; A C Subroutine For Parsing Command-Line Arguments; Arithmetic Aberations With dBASE-III and the dBASE Developer's Release; The Marriage of TEX and Postscript; REVIEWS: C Interpreters; C Source-Level Debuggers; Lomas Data Products' S-100 PC; CompuPro S-100 PC Video Board.

MAY/JUNE 1986 (Vol. 2, No. 3): Build An S-100 HD64180 CPU Card; Configuring MS-DOS; Writing MS-DOS Device Drivers—Part II; Building An AT Clone; Adding A RAM Disk To A CP/M System; Obfuscated C Code Contest; Sorting With Turbo Pascal; More dBASE Speed Techniques; Using an RCPM System; REVIEWS: Concurrent DOS; MEX-PC; ConIX; FirsTime.

JULY/AUGUST 1986 (Vol. 2, No. 4): Power to the PC; Creating A Copy Protected Program; Build Smart Keyboard Interface; Writing An MS-DOS Memory Disk Device Driver; Declare & Define C Variables in One File; More Loadable BIOS Drivers For CP/M; REVIEWS: Phoenix' PFIX-Plus Debugger; Holliston Challenger XT186; Magnum Digital PRO-180 & FD-100; XPIP; PMATE; Ryan-McFarland Fortran; WAT-COM; Watfor-77; MicroGlyph SciPlot; Softech Matrix Calculator; Personal TEX.

SEPTEMBER/OCTOBER 1986 (Vol. 2, No. 5): Control Systems Made Simple; Enhancing CP/M-80; A First Look At the 80386; Interfacing Using the SCSI Bus; Program Interfacing To MS-DOS—Part IX; Using SYMDEB With NMI Breakpoints; SYSLIB, Z3LIB, Z3LIB, & VLIB; REVIEWS: PC-Pro; CompuMagic Utility Package; Plu*Perfect Systems' DataStumper.

NOVEMBER/DECEMBER 1986 (Vol. 2, No.6): Roll Your Own PC Clone; Speeding Up the PC/XT; Program Interfacing to MS-DOS; Recovering PC-DOS Files; Cloning in the Fast Lane; MathSoft's MathCAD; Echelon's Z-SYSTEM.

MARCH/APRIL 1987 (Vol. 3, No.2): Multitasking Between Programs; Communicating at 9600-bps—Part II; Program Interfacing to Microsoft Windows—Part II; Classic Technology's 286 Speed Pak; Introducion of the LANScape column.

To Order: Return this Order Form with your payment to M & T Books, 501 Galveston Dr., Redwood City, CA 94063. *Or*, call TOLL-FREE (800) 533-4372 Mon-Fri 8AM-5PM. In CA call (800) 356-2002.

Issue	Price
	$
	$
	$
	$
	$
TOTAL	$

Name _____

Address _____

City _____ State _____ Zip _____

☐ Check Enclosed. **Make Payable to M & T Publishing Inc.**

Charge my ☐ VISA ☐ M/C ☐ Amer. Exp.

Card # _____ Exp.Date _____

Signature _____

The Software Directory

When contacting software publishers, please mention that you read about their product in Micro/Systems Journal.

Program Name: BLAST "Host"
Requirements: See description
Description: Links PCs to mainframes and minis. Runs under IBM's VM/CMS and MVS/TSO operating systems using any asynchronous pathways (e.g., NTO, protocol converters, and X.25 pads). Allows file transfer back and forth between systems. Can be used with MS-DOS, Macintosh, UNIX, VAX/VMS, Wang, Prime, and other operating systems.
Price: $5,500 (IBM mainframe); $495-$1,295 (minicomputers); $250 (PC or Macintosh).
Publisher: Communications Research Group, 5615 Corporate Blvd., Baton Rouge, LA 70808; (504) 923-0888.

Program Name: DISKGUISE
Requirements: CP/M 2.2
Description: Converts user areas into pseudo disk drives. Also creates a full-time "public drive." All user areas are immediately accessible by all application programs. Can retrieve any file and operates at all times, thus allowing programs such as WordStar to reside in one user area and yet be accessible everywhere in the system. Is customized to run only on licensed systems.
Price: $38 (includes 30-day, money-back guarantee)
Publisher: Logic Associates, 1433 W. Thome, Chicago, IL 60660; (312) 274-0531.

Program Name: Disk Maker II Plus
Description: High-speed disk-to-disk conversion for more than 350 floppy-disk formats. Program includes several MS-DOS, PC-DOS, CP/M, UNIX, and XENIX and several word-processor, minicomputer, and phototypesetter formats. It can handle 8-, 5¼- (360K, 720K, and 1.2MB), and 3½-inch disks. Options are available for IBM 374x series (System 34, 36, 38), autoloader applications, and disk alignment software.
Price: $5,995
Publisher: New Generation Systems, 2153 Golf Course Dr., Reston, VA 22091; (703) 471-5598.

Program Name: Hard Desk
Requirements: PC compatible
Description: Provides a user interface with a set of commands written in English and addressable with function keys for integrating as many as ten software packages under a single menu system.
Price: $95
Publisher: Modern Technologies International Inc., 656 Bair Island Rd., Redwood City, CA 94063; (415) 367-6855.

Program Name: Modula-2
Requirements: PC-DOS/MS-DOS system
Description: A Modula-2 compiler that supports the full language as defined in Niklaus Wirth's book *Programming in Modula-2*. Includes a UNIX-like make utility and the complete source code and object for the run-time system. No royalty is charged for the run-time object code. Source code is provided for the low-level PC-DOS interface coded in assembly language to demonstrate the interface between Modula-2 and assembly language.
Price: manual only, $25; complete package, $89.95.
Publisher: Farbware, 1329 Gregory, Wilmette, IL 60091; (312) 251-5310.

Program Name: Source Code Generator
Requirements: Turbo Pascal V3 for CP/M/Z-80
Description: A disassembler that generates the assembly code for the complete Pascal system (run-time library, editor, and compiler). Also fully documents the source file. Comes with a Z-80 assembler.
Price: $45 (Z-80 or CP/M V2.2); $75 (CP/M V3).
Publisher: C.C. Software, 1907 Alvarado Ave., Walnut Creek, CA 94586; (415) 939-8153.

Program Name: TransLISP PLUS
Requirements: PC-DOS/MS-DOS, minimum of 360K RAM
Description: A LISP development system. Includes an interface to Microsoft C and more than 400 Common LISP primitives. Allows users to customize LISP or combine C functions with LISP programs. Has support for Microsoft mouse and PC color graphics and is lexically scoped. Has a trace facility, pretty printer, and cross-reference for variables and functions, and users can invoke DOS functions and applications from the system. On-line help is provided.
Price: $195 (includes tech support and 30-day, money-back guarantee); run-time interpreter, $150.
Publisher: Solutions Systems, 335 Washington St., Norwell, MA 02061; (800) 821-2492 or (617) 659-1571.

Program Name: Tree Diagrammer
Description: Clarifies program source code in C, BASIC, Pascal, dBASE, and Modula-2. Automatically makes an organization chart of a program's subroutine structure. Shows hierarchy of calls to procedures, functions, and subroutines. Recursive calls are indicated. Specially flagged comments within the source code will appear on the chart.
Price: $55
Publisher: Aldebaran Laboratories Inc., 3339 Vincent Rd., Pleasant Hill, CA 94523; (415) 930-9866.

Program Name: VERSBASE
Requirements: CP/M 2.2
Description: Stamps version numbers on files. When a selected file is to be renamed to .BAK program, asks users if file should be given a version number instead. If yes, next highest version number for that file is appended to file's forename—for example, TEST.ASM becomes TEST0001.ASM.
Price: $38 (includes 30-day, money-back guarantee)
Publisher: Logic Associates, 1433 W. Thome, Chicago, IL 60660; (312) 274-0531. §

Classifieds

Micro/Systems Journal *accepts Classified Ads. The charge is $6/line (3 lines minimum); 40 characters max./line. Three times frequency $15/line; six times $25/line; non-profit clubs $2/line. Logos, special type, etc. are extra charge. Check must accompany ad copy. Send to M & T Publishing Inc., 501 Galveston Dr., Redwood City, CA 94063.*

DOS User's Group
Quarterly Newsletter discounts on Third Party Products. Latest DOS news demo diskettes. DOS Tutorials. Annual membership fee: U.S. $25, Canada & Europe $35. DOS User's Group, P.O. Box 26601, Las Vegas, NV 89126.

ENGINEERING SOFTWARE
Circuit Design and Analysis Programs. FREE CATALOG and TUTORIAL GUIDE. BV Engineering, 2200 Business Way, #207, Rverside, CA 92501; (714) 781-0252.

RAM DISK
S-100, 2 MEG, PORT I/O. New, Warranteed, $725. S. Lugert, 439 Peck Slip Sta., N.Y.C., N.Y. 10272, or call (718) 622-0654.

Sunol Systems, Pleasanton, California, announced an agreement with **Digital Services Corp.**, Schaumburg, Illinois, to provide maintenance services for Sunol's line of hard disk drive file servers and disk servers. Digital Services will provide both Warranty and Non-Warranty service for the complete line of Sunol products. Call (312) 351-2266 for further info.

Run CP/M Programs on IBM PC With EZ80! V-20 or Plug-in Board NOT REQUIRED. Includes terminal emulation for: Wyse 50, H19, ADM3A, Osborne, Kaypro. Type CP/M filename and press RETURN! $40.00, tax & shipping included. HG Software Co., PO Box 10151, Austin, TX 78766-0151; (512) 836-3188.

POOR MAN'S NETWORK
A true local area network that lets you share resources between 2 CP/M machines, using standard RS-232 or parallel ports. Works with CP/M 2.2, ZCPRx, Echelon's ZRDOS, Micro Methods' RP/M. Only $69 (U.S.). For info, call (613) 722-0690 Tues-Sat 10AM-5PM or write to Anderson Techno-Products Inc., 947 Richmond Road Dept. S, Ottawa, Ontario K2B 6RI, Canada.

NO SOURCE CODE?
REL/PAK converts Microsoft REL files to 8080 and Z80 source code MAC files including symbols under CP/M-80 and TurboDOS. Now $99.95 on 8″ SSSD. Microsmith Computer Technology, POB 1473, Elkhart, IN 46515. (800) 622-4070 or in IL (800) 942-7317.

Unix for the programmer: VALID SG-10 PC: (Corvus) 68010 cpu, 20MB hd, floppy, 2MB RAM, mouse, kbd, smart monitor. New, running. Unix boots on hd. $1000. Mike Kotlan, (503) 596-2050.

NEW! 68000 COMPUTERS w/software $395. 68000 Operating System w/source $50. **HAWTHORNE TECHNOLOGY**, 8836 SE Stark, Portland, OR 97216; (503) 254-2005.

HSC MC68000 board for IBM PC. Sep. OS runs in parallel w. MS-DOS. Inc: OS9 w. lang pkg (C, Pas, and Bas) and CPM68k with C comp. 10MHz upgr. to 12.5, has NS32081 math coproc. + lib. 1MB RAM cap. Can be upgr. to 680xx at factory. Good system for sci/eng or CD ROM dev. under OS9. $580. Bill (617) 926-0606.

COMPUTER PROFESSIONALS
with good writing skills to evaluate & review latest computer software products for national sydication. Part-time, contract stringer. Must have access to IBM PC, MAC, or compatible clones. Access to DEC equipment and some CPM PC. Send resume: Software Programmers News PSPA, 830 Bay Ave. #G, Dept 7, Capitola, CA 95010.

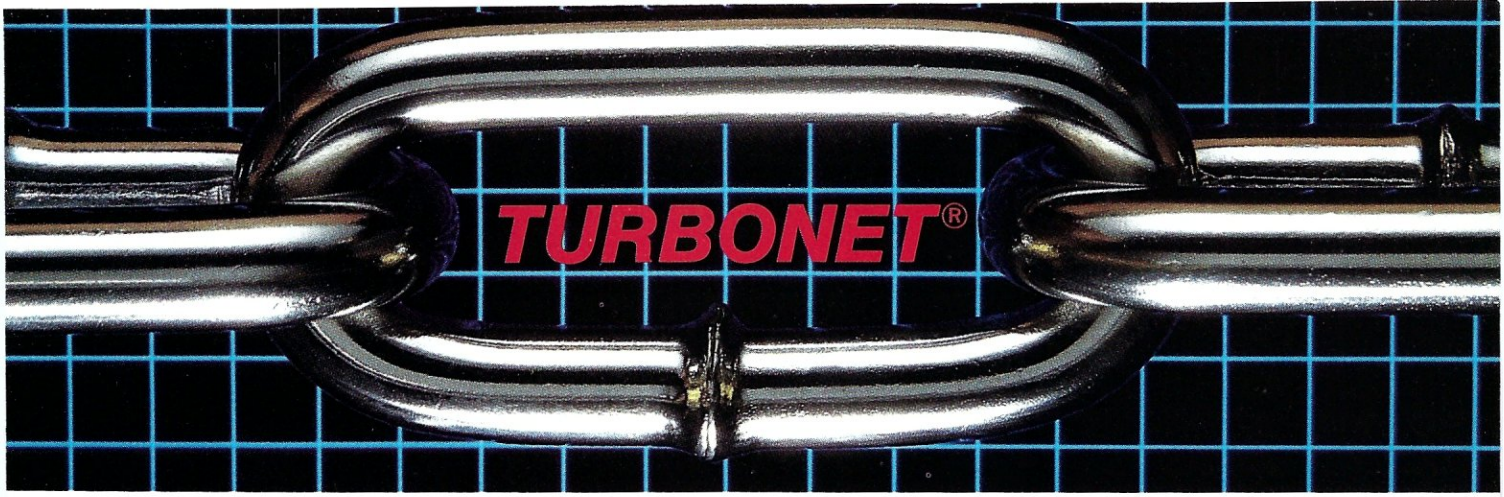
IBEX BUS SYSTEMS, CP/M, 2-1.2MB, 8″ flpy, b/w Mon., Wordstar, Calcstar, Turbo Pascal & more. Like new. Cost $3500. Make offer. John Anderson (805) 496-6408, (213) 336-7573.

FOR SALE: TurboDOS/S-100 w/4-TV-912 terminals, 90MB micropolis drive, 2-8″ drives + CP/M software. $3,500/offer. (800) 992-0412.

DISK SERVICE MANUAL ($20)
"Printer/Plotter Manual" ($15); "Super Re-Inking Method" ($7); "Computer Phreaking" ($15); "Absolute Computer Security" ($15). 40+ computer & electroics manuals & software, Catalog: $1. By former NMSU CS Professor. CONSUMERTRONICS, 2011 Crescent, Alamogordo, NM 88310.

Advertiser Index

Anderson Techno-Products	80
Andy Bakkers	73
Austin Code Works	55
BG Computer Applications	53
BV Engineering	80
C Users Group	73
California Digital	41
Charles M. Kotlan	80
Clary Corp.	7
Cobalt Blue	55
Competitive Edge	9
CompuMagic Inc.	43
CompuPro/Viasyn	1
Consumertronics	80
Custom Software Systems	5
D & W Digital Inc.	21
Datalight	13
Digital Decisions	69
Digital Research Computers	29
Digital Services Corp.	80
DOS User's Group	80
Dr. Dobb's Subscriptions	24
Echelon Inc.	39
Ecosoft, Inc.	26
Exec-PC Inc.	31
Gimpel	15
Gimpel	63
Hawkeye Grafix Inc.	61
Hawthorne Technology	80
HG Software Co.	80
Integrand Research Corp.	51
John Anderson	80
Laboratory Microsystems Inc.	49
Lahey Computer Systems Inc.	69
Leadtrack	80
Lodden Technology Ltd.	20
Lomas Data Products	45
M & T Books	33,71,77,78
Macrotech International	C-2
MetaWare Inc.	27
MetaWare Inc.	61
Micromint	18
Microsmith Computer Technology	80
Micro/Systems Subscriptions	56
Night Owl	44
Periscope Co. Inc.	40
Personal Business Solutions	64
Programmer's Journal	47
Qualstar Corp.	35
Quarterdeck Office Supplies	2
Raima Corp.	17
S. Lugert	80
SemiDisk Systems	C-4
Sharpe Systems	11
SLR Systems	35
Slicer Computer	64
Software Programmer's News	80
Solution Systems	37
Sunny Hill Software	5
Teletek	C-3
Western Ware	69
William van Riper	80



One Strong Link Can Strengthen Your Whole System

TurboNET® is Teletek's new local area network that permits reliable high speed data transfers among computers of differing bus architectures. TurboNET is designed to be used in networks consisting of multiple S-100 based and PC based systems. It will allow up to 4000 users, including up to 255 IBM PCs or compatibles, to share a single network and all attached peripherals. The network can be organized in any number of different ways mixing Teletek's 8 and 16-bit multiuser systems and PCs in any combination.

Teletek's Networking Family consists of:

TurboNET PC: IBM-PC Network Interface Board

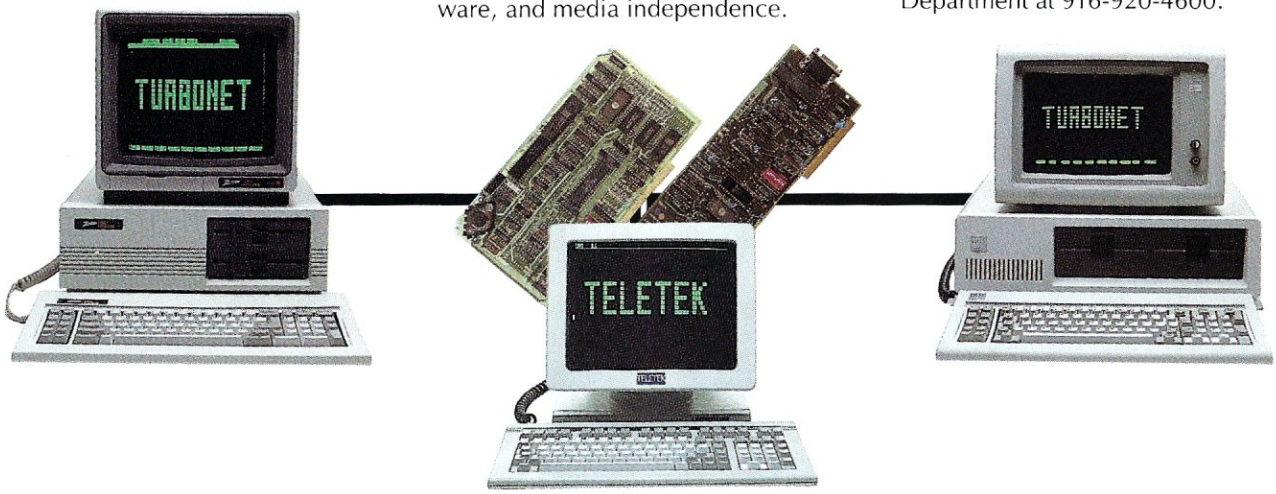
Teletek's TurboNET PC board offers IBM-PC Compatibility, CSMA industry standard protocols, 2 megabits/sec network speed, on-board CPU and communications management firmware, and media independence.

TurboNET S-100: S-100 Network Interface Board

Teletek's TurboNET S-100 board offers IEEE 696 Compatibility, CSMA industry standard protocols, 2 megabits/sec network speed, on-board CPU and communications management firmware, and media independence.

The benefits are obvious: The cost savings of shared peripherals, almost unlimited system expansion capability, and the use of existing PC workstations with the ability to run the myriad of application software written for MS-DOS and PC-DOS. This coupled with Teletek's 8 and 16-bit multiuser systems running application software written for CP/M and MP/M allows the system the ability to access almost any software library.

For more information on Teletek's TurboNET S-100 and TurboNET PC boards or on any of our full line of S-100 products, please call our Sales Department at 916-920-4600.



TELETEK

4600 Pell Drive, Sacramento, CA 95838
(916) 920-4600 Telex #4991834 Answer back — Teletek

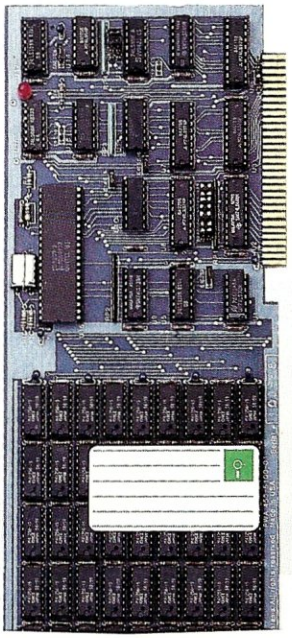
© 1986 Teletek

Think fast! Pick the better fit...

Our 5th Year Bonus!
Mention this ad when ordering
and get your choice of a V 20-8
(replaces 8088) or $20 off on
your Battery Backup purchase!

FLOPPY DISK.

- Fills time between coffee breaks
- Makes a hard disk seem *fast*
- Your computer appears busy (even if you aren't!)
- Wears out moving parts



SEMIDISK Disk Emulator.

- Gets that job done NOW
- Makes a hard disk seem *slow*
- Maximizes your productivity with anything from databases to compilers
- Totally silent operation

...for YOUR demanding tasks.

SURPRISE! *Neither* is memory mapped, so they don't affect your precious Main Memory. *Both* retain data indefinitely - even with the computer turned off.

THE SEMIDISK SOLUTION. You could invest in a series of "upgrades" that turn out to be expensive band-aids without solving your real problem. Even those "Accelerator" and "Turbo" boards do little to speed up disk-bound computers. If your applications spend too much time reading and writing to disk (and whose don't?), you won't want to settle for anything less than a SemiDisk disk emulator. The SemiDisk comes in 512K and 2Mb capacity. More boards may be added to make up to an 8 Megabyte SemiDrive!

SPEED THAT'S COMPATIBLE. PC, XT or AT, if you need speed, the SemiDisk has it. How fast? Recent benchmarks show the SemiDisk is from 2 to 5 times faster than hard disks, and from 25% faster (writing) to several times faster (random reads) than VDISK and other RAMdisk software that gobble up your main memory.

MEMORY THAT'S STORAGE. Using our small external power supply, with battery backup, your data remains intact through your longest vacation or even a seven-hour power failure!

CELEBRATE WITH US! Now, SemiDisk celebrates its fifth birthday with a special offer for IBM-PC owners. Buy a SemiDisk now and we'll include an 8 MHz V-20 microprocessor (replaces the 8088) to make your new SemiDisk run even faster. Don't need the V-20? We'll take $20 off the price of your Battery Backup Unit!

	512K	2Mbyte
IBM, PC, XT, AT	**$495**	**$ 795**
Epson QX-10	**$495**	**$ 995**
S-100 SemiDisk II	**$795**	**$1295**
S-100 SemiDisk I	**$299**	**----------**
TRS-80 II, 12, 16	**$495**	**$ 995**
Battery Backup	**$130**	**$ 130**

Someday you'll get a SemiDisk.
Until then, you'll just have to...wait.

SemiDisk

MasterCard VISA

SemiDisk Systems, Inc., 11080 S.W. Allen Blvd., Beaverton, Oregon 97005 (503) 626-3104